Date / /

Graphics (graphics.h) - c programming.

Graphics programming in c used to draw various geometrical shapes (rectangle, circle, ellipse, etc), use of mathematical functions in drawing curves, coloring an object with different colors and patterns and simple animation programs like jumping ball & moving cars.

C graphics using graphics.h functions can be used to draw different shapes, display text in different fonts, change colors and many more. Using functions of graphics.h in Turbo c compiler or any c compiler you can make graphics programs, animations, projects and games. You can draw many geometrical figures. You can change their colors using the available functions and fill them.

→ **Functions in graphics.h**

1) Arc function in c.

Declaration : void arc (int x, int y, int stangle, int endangle, int radius);

Used to draw an arc with center (x,y) and stangle specifies starting angle,

Date / /

endangled specifies the end angle and last parameter specifies the radius of the arc. It can also be used to draw a circle but for that starting angle and end angle should be 0 and 360 respectively.

2) Bar function in c -
Declaration -
void bar (int left, int top, int right, int bottom);

Used to draw a 2-dimensional rectangular filled in bar.

3) Circle -
Declaration -
void circle (int x, int y, int radius);
Used to draw a circle with center (x,y) and third parameter specifies the radius of the circle.

4) Cleardevice -
Declaration -
void cleardevice();
clears the screen in graphics mode and sets the current position to (0,0). consists of filling the screen with current bg color.

Date / /

5) closegraph -
Declaration -
void closegraph();
closes the graphics mode, deallocates all memory allocated by graphics system and restores the screen to the mode it was in before you called initgraph.

6) drawpoly -
declaration -
void drawpoly (int num, int *polypoints);

num → (n+1) number of points where n is the no. of vertices in a polygon. polypoints points to a sequence of (n*2) integers.
Draw polygons like triangle, rectangle, pentagon, hexagon, etc.

7) ellipse -
Declaration -
Void ellipse (int x, int y, int stangle, int endangle, int xradius, int yradius);

8) fillellipse -
Declaration -
void fillellipse (int x, int y, int xradius, int yradius);

9) fillpoly - draws & fills the polygon.

void drawpoly (int num, int *polypoints);

Date / /

(10) getcolor
Returns the current drawing color.
Ref: int getcolor();
eg- a = getcolor();

(11) line-
void line (int x1, int y1, int x2, int y2);

(11) putpixel-
void putpixel (int x, int y, int color);
eg- putpixel (35, 35, GREEN);

(12) textwidth
(13) textheight
(14) setviewport
(15) settextstyle
(16) setlinestyle
(17) setfillstyle
(18) setcolor
(19) setbkcolor
(20) sector
(21) rectangle
~~(22) putpixel~~
(23) pieslice
(24) outtextxy
(25) outtext
(26) moverel
(27) moveto
(28) linerel
(29) lineto
(30) imagesize
(31) graphicserrormsg
(32) graphdefaults
(33) gety
(34) getx
(35) getpixel
(36) getmaxy
(37) getmaxx
(38) getmaxcolor
(39) getimage
(40) getdrivername
(41) getbkcolor
(42) getaxcords
(43) floodfill
(44) bar3d

Date / /

→ Steps to install or Add graphics.h in Dev-C++;

By using Graphics library, we can draw lines, Rectangles, ovals, Arcs, polygons, images, and strings on a Graphical window.

1) Downloading Dev C++ (C compiler)-
- Download Dev C++ from its official website or Sourceforge.
- Install DevC++.

2) Adding Graphics Header File in Dev C++.
- Download required files.
- Unzip the downloaded files.
- Copy "graphics.h" and "winbgim.h" files to "include" folder of Dev C++ Directory.
- Copy "libbgi.a" to "lib" folder of Dev C++ directory.
- Now open Dev C++, File => New => project => Basic => Console Application.
- Go to project => project options [or Ctrl+H].
- Select parameter tab from project options.
In linker add following codes.
- lbgi
- lgdi32
- lcomdlg32
- luuid
- loleaut32
- lole32.
- click OK

Date / /

- Select TDM-GCC x.x.x 32-bit Release from Drop down menu.
- All done, now check whether all steps followed correctly or not by running a small graphics program.
- So, our graphics program running successfully.

→ Write a c program to color a pixel x, y.

```
#include<graphics.h>
#include<conio.h>

main()
{
  int gd = DETECT, gm;
  initgraph(&gd, &gm, "C:\\TC\\BGI");
  putpixel(25, 25, RED);
  getch;
  closegraph();
  return 0;
}
```

* Output of this program will be a red pixel on screen at (25,25). Try to spot that pixel with your eyes at left top portion of your computer screen.

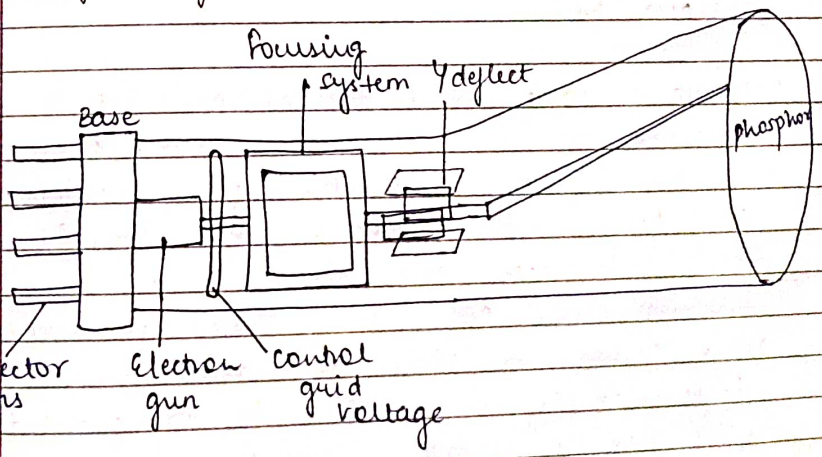
Date / /

→ Cathode Ray Tube -

CRT stands for Cathode Ray Tube (CRT) is a technology used in traditional computer monitors and televisions. The image on CRT display is created by firing electrons from the back of the tube of phosphorus located towards the front of the screen.

Once the electron heats the phosphorus, they light up, and they are projected on a screen. The color you view on the screen is produced by a blend of red, blue and green light.

Diagram of CRT -



Electron - subatomic particles that ...
elementary charge of magnitude -1.
mass of electron = 9.1×10^{-31} kg. Rated 0.00055u ...

→ Components of CRT:

Main components of CRT are:

1) Electron Gun - consisting of a series of elements, primarily a heating filament (heater) and a cathode. The electron gun creates a source of electrons which are focused into a narrow beam directed at the face of the CRT.

2) Control Electrode - Used to turn the electron beam on and off.

3) Focusing system - Used to create a clear picture by focusing the electrons into a narrow beam.

4) Deflection Yoke: Used to control the direction of the electron beam. It creates an electric or magnetic field which will bend the electron beam as it passes through the area. In a conventional CRT, the yoke is linked to a sweep or scan generator. The deflection yoke which is connected to the sweep generator creates a fluctuating electric or magnetic potential.

Electrically neutral atoms/molecules must have an equal no. of electrons & protons. Date / / protons, neutron, electr...

5) phosphorous-coated screen -
The inside front surface of every CRT is located coated with phosphors. phosphors glow when a high energy electron beam hits them. phosphorescence is the term used to characterize the light given off by a phosphor after it has been exposed to an electron beam.

* An electrical charge is created when electrons are transferred to or removed from an object. Because electrons have a negative charge, when they are added to an object, it becomes negatively charged. When electrons are removed from an object, it becomes positively charged.

* cathode is the place on a battery or other electrical device where the electric current leaves.

* Cathode ray tube is a vaccum tube containing one or more electron guns, which emit electrons beams that are manipulated to display images on a phosphorescent screen. The images may represent electrical waveforms, pictures, radar targets, or other phenomena.
Also called as a picture tube.

Date / /

→ Video display device.
display device is an output device for presentation of information.

→ Types of video display devices available.

→ Terminology related to display devices -
pixel -
Resolution -
Aspect ratio -
Refreshing -

→ Diff. b/w Raster and Random Scan Displays.

→ Short Note on eq^{n} of line.
The equation of a straight line is
$y = mx + c$
m is the gradient
c is the height at which the line crosses
y-axis also known as the y-intercept.

→ Short note on eq^{n} of circle.
general equation of a circle is
$(x-h)^2 + (y-k)^2 = r^2$
(h,k) is the center and r is the radius.

Date / /

→ Short note on eq^{n} of ellipse.
The standard eq^{n} of an ellipse is
$x^2a^2 + y^2b^2 = 1 \times 2a^2 + y^2b^2 = 1$,
where a is the length of semi-major axis and b is the length of semi-minor axis.

Q. Consider 3 different raster system with resolution -
1) 640×480
2) 1280 × 1024
3) 2560 × 2048.
find the size of frame buffer in system to store 12 bits / pixel.

1) $\frac{640 \times 480 \times 12 bit}{8 \times 1024} = 450 Kb.$

2) $\frac{1280 \times 1024 \times 12}{8 \times 1024} = 1920 Kb$

3) $\frac{2560 \times 2048 \times 12}{8 \times 1024} = 7680 Kb.$

Q. Find the aspect ratio for all 3 systems -

1) $\frac{640}{480} = 4:3$

2) $\frac{1280}{1024} = 5:4$

3) $\frac{2560}{2048} = 5:4$

Date / /

Aspect Ratio-
It is the ratio of an image of its width to its height for x:y, the image is x unit y and y unit x.

The most common aspect ratio is 4:3 for traditional television and computer monitors.

For modern smart phones 6:13 aspect ratio is common.

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

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

Q. Consider 2 Raster systems with resolution 1) 640 x 480 and 2) 1280 x 1024. How many pixels could be accessed per sec in each of these systems by display controller that refreshes the screen at a rate of 60 fps.

Date / /

1) 1 frame → 640 x 480 pixels
60 fps → 60 frame → 1 sec
Total frames in 1 sec → 640 x 480 x 60 pps

2) Total frame in 1 sec → 1280 x 1024 x 120 pps.

fps => Resolution x Refresh Rate.

Resolution - The max. number of points that can be displayed without overlapping in the screen is refers as resolution.

Q. How much time is spent scanning across each row of pixel and each pixel on a raster system with resolution of 1280 x 1024 and the refresh rate of 60 fps.

no. of rows → 1024
no. of pixel in 1 row → 1280.

1) 1280 x 1024 → resolution
(rows)
1024 x 60 → 1 sec
1 pixel → $\frac{1}{60 \times 1024}$ = 0.058 sec.

2) 1280 x 1024 x 60 → 1 sec
1 pixel → $\frac{1}{60 \times 1024 \times 1280}$.

Date / /

→ **Line Drawing Algorithm -**

In computer graphics line drawing algorithm is a technique for approximating a line segment on pixel based display and printers.

There are two basic algorithms -

1) DDA line drawing Algorithm.
(Digital Differential Analyser).

Equation of line is $y = mx + c$

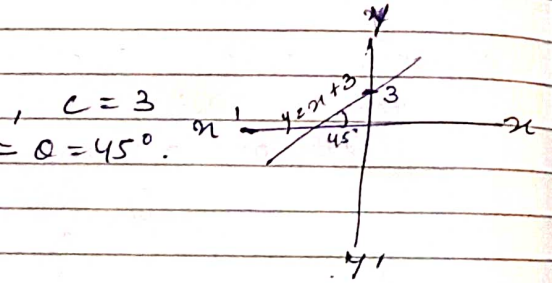
where m is the slope
c is the intercept.

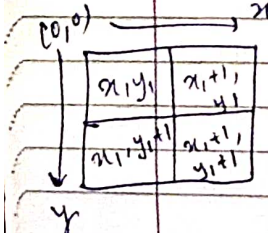
$m = \tan\theta$.

eg: $y = x + 3$

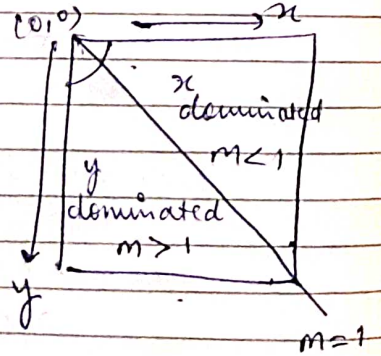
$m = 1$, $c = 3$

$\tan\theta = \theta = 45°$.





$$\text{Slope} = m = \frac{y_2 - y_1}{x_2 - x_1}$$



Case 1: $m < 1$

Case 2: $m > 1$

Case 3: $m = 1$

Date / /

Case 1 - $m \leq 1$. → when slope is < 1.

$$m = \frac{y_2 - y_1}{x_2 - x_1} = \frac{dy}{dx} = \frac{\Delta y}{\Delta x}$$

$$\frac{dy}{dx} < 1$$

$$\boxed{dy < dx}$$

So, maximum increment is given to x-value which is of one unit. and we need to calculate the change in y.

$$\boxed{dx = 1}$$

$$m = \frac{dy}{dx}$$

$$m = \frac{dy}{1} \Rightarrow \boxed{dy = m}$$

$$\therefore \boxed{\begin{aligned} x &= x + 1 \\ y &= y + m \end{aligned}}$$

→ decimal value.

Case 2 - $m > 1$ → when slope is > 1.

$$\frac{dy}{dx} > 1 \Rightarrow \boxed{dy > dx}$$

So, max. change of one unit will be given to y. and we need to calculate the change in x.

so, $\boxed{dy = 1}$

$$m = \frac{dy}{dx}$$

$$m = \frac{1}{dx} \Rightarrow \boxed{dx = \frac{1}{m}}$$

$$\therefore \boxed{\begin{aligned} x &= x + \frac{1}{m} \\ y &= y + 1 \end{aligned}}$$

Date / /

Q. Find the pixel values of the line segment joining points / pixels (10, 12) and (20, 21).

$$\frac{m = y_2 - y_1}{x_2 - x_1} = \frac{21-12}{20-10} = \frac{9}{10} = 0.9$$

↙ no. of pixels.

Q1) What do you mean by computer graphics?

Ans) The branch of science and technology concerned with methods and techniques for converting data to or from visual presentation using computers.

Q2) What are the applications of computer graphics?

Ans) 1) Computer Aided Design
2) Graphical user interface
3) Entertainment
4) Simulation and Training
5) Education and Presentation
6) Computer Generated Art
7) Scientific Visualization
8) Image Processing
9) Virtual Reality
10) Cartography.

Q3. What do you mean by "interactive Computer graphics?

Date / /

Ans) Interactive computer graphics like a website, it is only useful if it is browsed by a visitor and no two visitors are exactly alike. It means the website must support the interaction of users with a variety of skills, interests and end goals. Interactive computer graphics involves the user's interaction.

Q4) What do you mean by GUI?

It stands for graphical user interface. A major computer of a GUI is a window ~~manajor~~ manager that allows a user to display multiple-window areas. To make a particular window active we simply click in that window using an interactive pointing device. Interfaces also display menus and icons for fast selection of processing options or parameter values.

Q5) What does it mean by RGB?

The RGB is a color model, it is an additive color model in which red, green and blue light are added together in various ways to reproduce a broad array of colors. The name of the model comes from the initials of the three aditive primary colors, red, green and blue. The main purpose of the RGB color model is for the sensing, representation, and display of images in electronics systems, such as television & computers though it also been used in conventional

Date / /

Q6) Define VDU?
A monitor or display (sometimes called visual display unit) is a piece of electrical equipment which displays images generated by devices such as computers, without producing a permanent record. The monitor comprises the display device, circuitry, and an enclosure. The display device in modern monitors is typically a thin film transistor liquid crystal display, while older monitors uses a cathode Ray tube.

Q7) Define persistence in terms of CRT phosphorous?
Persistence is the one of the major property of phosphorous used in CRT's. It means how long they continue to emit light after the electron beam is removed.

Q8) Define resolution?
Maximum number of points that can be displayed without overlap on a CRT is refered to as the resolution.

Q9) What do you mean by an aspect ratio?
Aspect ratio is a no. which gives the ratio of vertical points to horizontal points necessary to produce equal length lines in both directions on the screen. An aspect ratio of 3/4 means that

Date / /

a vertical lines plotted with three points has same length as a horizontal line plotted with 4 points.

Q10) What are the different properties of phosphorous?
1) color
2) Persistence.

Q11) Difference raster and random scan displays?
In a raster scan displays the electron beam is swept across the screen, one row at a time from top to bottom. Contrasting in random scan displays the electron beam is directed to the parts of the screen where a picture is to be drawn.

Q12) Define refresh buffer / frame buffer?
Picture definition is stored in a memory area called the refresh buffer or frame buffer. This memory area holds the set of intensity values for all the screen points.

Q13) Define Pixel.
Each screen point is refered to as a pixel or pel.

LAB ASSIGNMENT - 3

Date / /

Q1) Write a program to implement dda line drawing algorithm.

Q2) Write a program to check if a point lies on a line or not? Coordinates of line (20,20), (50,50) and the point coordinates are (30,30).

Sol1):

```
#include <graphics.h>
#include <conio.h>
#include <stdio.h>
void main()
{
int gd = DETECT, gm, i;
float x, y, dx, dy, steps;
int x0, x1, y0, y1;
initgraph (&gd, &gm, "C:\\TC\\BGI");
setbkcolor (WHITE);
x0=100, y0=200, x1=500, y1=300;
dx=(float) (x1-x0);
dy=(float) (y1-y0);
if (dx>=dy)
  {
   steps = dx;
  }
else
  {
  steps=dy;
}
dx=dx/steps;
dy=dy/steps;
```

Date / /

```
x=x0;
y=y0;
i=1;
while (i<=steps)
{
putpixel (x,y,RED);
x+=dx;
y+=dy;
i=i+1;
}
getch();
closegraph();
}
```

Date / /

→ Filling Algorithm -

Process to fill the colour in close area

seed-pixel → from where the filling starts.

There are two basic algorithms for filling the colour.

1) Boundary fill
2) Flood fill

→ Boundary fill algorithm -

In this method color is filled in the interior area until the boundary is reached. The color of current pixel is checked with boundary colour and fill colour if pixel is not coloured with fill colour then it will be coloured otherwise not.

The algorithm is as follows:

Step 1: Input the ~~seed~~ boundary values, seed pixel and two colors which are fill color and boundary color.

Step 2: Check the color of current pixel and check if it is equal to fill colour and boundary colour.

Date / /

if yes, go to stop.
Otherwise go to step 3.

Step 3: Fill the current pixel with fill colour and recursively call 4-connected or 8-connected model for finding neighbouring pixels.

Step 4: stop.

function -

```
void Boundary_fill(int x, int y, int fillcolor,
                   int boundarycolor)
{
    int c = getcolor(x,y);

if (((c != fillcolor) && (c != boundarycolor))
{
    putpixel (x,y, fillcolor); delay(100);
    Boundary-fill (x+1, y, fillcolor, boundcolor);
                  (x, y+1, fillcolor, bound.color);
                  (x, y-1, fillcolor, bon.color);
                  (x-1, y, fillcolor, boundcolor);
}

main()
{
 circle (100,100,50)   RED BLACK.
 Boundary fill (100,100, Red, Black);
```

Date / /

→ Flood fill algorithm -

The limitation of boundary fill algorithm is it says when the boundary is of multiple colour, in this case flood fill algorithm is used which works on the idea of interior colour or original colour. The interior colour of the object or filling area is replaced with the fill colour and this process continues until all the pixels of interior area are coloured.

The algorithm is as follows -

Step1 - Initialise the seed point, area to colour, fill colour and interior colour.

Step2 - Check if the colour of current pixel is equal to interior colour then go to step 3 otherwise go to step 4.

Step3 - Colour the current pixel with fill colour and repeat the process using 4-connected or 8-connected model.

Step 4 - Stop.

Date / /

```
void floodfill (int x, int y, int fillcolor, inte
 intcolor)
{
  if (getcolor (x,y) == intcolor)
  {
    putpixel (x, y, fillcolor);
    floodfill (x+1, y, fillcolor, intcolor);
    floodfill (x, y+1, fillcolor, intcolor);
    floodfill (x-1, y, fillcolor, intcolor);
    floodfill (x, y-1, fillcolor, intcolor);
  }
```

Lab Assignment -

1) Boundary fill
2) Flood fill.
3) point clipping.