

Graphics in C Language

Concept of Computer Graphics:

- Computer Graphics is one of the most powerful and interesting aspect of computers.
- Graphics programming are used to drawing various geometrical shapes(rectangle, circle, line, ellipse etc.), use of mathematical function in drawing curves, coloring an object with different colors and patterns and simple animation programs.

Concept of C Graphics:

- There is a large number of functions in C which are used for putting pixel on a graphic screen to form lines, shapes and patterns.
- The default output mode of C language programs is text mode. So we need to switch into graphic mode before drawing any graphical shape like line, rectangle, circle, ellipse etc.

Graphics In C:

- At first, we have to include `<graphics.h>` as the header file.

Initializing Graphics Mode:

- The `initgraph ()` function is used to switch the output screen from text mode to graphics mode.
- This function takes three arguments.
- **Syntax:**

```
initgraph(&Graphics_Driver, &Graphics_Mode, Path_To_Driver);
```

Graphics Function in C:

- There are many graphics library functions in C language which are used to draw different drawing shapes.

- Some of them are:

1. `line(x1,y1,x2,y2);`
2. `arc(xCenter,yCenter,start,end,radius);`
3. `circle(xCenter,yCenter,radius);`
4. `ellipse(xCenter,yCenter,start,end,Xradius,Yradius);`
5. `floodfill(x1,y1,color);`
6. `rectangle(x1,y1,x2,y2);`
7. `getmaxx();`
8. `getmaxy();`
9. `closegraph();`

1. line():

- This function is used to draw a line from a point (x_1, y_1) to point (x_2, y_2) i.e. (x_1, y_1) and (x_2, y_2) are end points of the line.
- In simple, this function is used to draw a line between two specified points.

Syntax:

line(x1,y1,x2,y2);

Example of line () Function:

```
#include<stdio.h>
#include<conio.h>
#include<graphics.h>
int main()
{
    int gd=DETECT, gm;
    initgraph(&gd,&gm, "C:\\TURBOC3\\BGI");
    setcolor(GREEN);
    line(50,50,500,600);
    getch();
    closegraph();
    return 0;
}
```

2. arc():

- The arc() function is used to draw an arc with specified center (x,y), starting angle, end angle and radius of the arc.
- This function is also used to draw a circle but for that starting angle and end angle should be 0 and 360 respectively.
- Syntax:

arc(xCenter,yCenter,start,end,radius);

3. circle() :

- This function is used to draw a circle with center (x,y) and third parameter specifies the radius of the circle.
- Syntax: **circle(xCenter,yCenter,radius);**

4. ellipse():

- This function is used to draw an ellipse.
- Syntax: **ellipse(xCenter, yCenter, start, end, Xradius, Yradius);**

Where, (x,y) are the coordinate of center of the ellipse, start is the starting angle, end is the ending angle, Xradius and Yradius specifies the radius of the ellipse.

5. floodfill():

- This function is used to fill an enclosed area.
- Syntax: **floodfill(x1,y1,color);**
- Example: `floodfill(300,300,RED);`

6. rectangle():

- This function is used to draw a rectangle.
- Syntax: **rectangle(x1,y1,x2,y2);**

Where, (x1,y1) is the coordinate of the left top and (x2,y2) is the coordinate of the right bottom.

7. getmaxx():

- This function is used to returns the maximum X coordinate for current graphics mode and driver.
- Syntax: `getmaxx()`;
- Example: `int xmax=getmaxx()`;

8. getmaxy():

- This function is used to returns the maximum Y coordinate for current graphics mode and driver.
- Syntax: `getmaxy()`;
- Example: `int ymax=getmaxy()`;