Paper 102: Programming & Problem solving through C

Lecture-30:Graphics

Filling & Patterns

- fillpoly() function allows filling the inside of polygon with fill pattern or fill color,

```
void far fillpoly(int number,int far *addrList)
```

– number	number of points
– addrList	address of list of points

- The fill pattern and fill color are set using setfillstyle() function

```
void far setfillstyle(int pattern,int color)
```

Filling & Patterns, Cont’d

Name	Number	Result
EMPTY_FILL	0	Solid Background
SOILD_FILL	1	Solid Color
LINE_FILL	2	Horizontal lines
LTSLASH_FILL	3	Thin Lines
SLASH_FILL	4	Thick Lines
BKSLASH_FILL	5	Thick Lines
LTBKSLASH_FILL	6	Thin Lines
HATCH_FILL	7	Light Hatch

Filling & Patterns, Cont’d

Name	Number	Result
XHATCH_FILL	8	Heavy Cross Hatch
INTERLEAVE_FILL	9	Interleaved Lines
WIDE_DOT_FILL	10	Wide spread dots
CLOSE_DOT_FILL	11	Close spaced dots
USER_FILL	12	User defined pattern

- Another approach is to fill the area with floodfill() function

Example – Using fill/pattern

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

void main(void)
{
  int pane[]={150,50, 180,20, 180,120, 150,150, 150,50};
  int driver=DETECT, mode;
  initgraph(&driver, &mode,“c:\\tc\\bgi");
  setfillstyle(SOLID_FILL, GREEN);
  setcolor(GREEN);
  fillpoly(5, pane);
  getch();
  closegraph();
}
```

Relative Positioning

- Instead of fixed coordinate system, several graphics functions use relative coordinate system
- Lines are only graphics element that can be drawn using relative coordinate system
- Drawing is done relative to a movable point called *current position* (CP)

```
void far linerel(int dx, int dy)
```

  - dx is horizontal distance from CP
  - dy is the vertical distance from CP

- To move the CP by absolute values

```
void far moveto(int x, int y)
```

where x and y are absolute coordinates

```
#include<graphics.h>
#include<conio.h>
#define MAX 320
#define GRID 40
#define SIDE 36
void square(int side);
void main(void){
    int driver=DETECT, mode;
    int x, y;
    initgraph(&driver, &mode,“c:\\tc\\bgi");
    for(y=0; y<MAX; y+=GRID)
          for(x=0; x<MAX; x+=GRID){
                    moveto(x,y);
                    square(SIDE);
          }
    getch();
    closegraph();
}
void square(int side){
    linerel(side, 0);         //top, left to right
    linerel(0, side);         //right side, top to bottom
    linerel(-side,0);   //bottom, right to left
    linerel(0,-side);         //left side, bottom to top
}
```

Drawing an Arc

- arc() function allows us to draw arc

```
void far arc(int xA, int yA, int stAngle,
                      int endAngle,int Radius)
```

- where

xA, yA	center of arc
stAngle, endAngle	starting and ending angles
radius	radius of arc

Example- Drawing an Arc

```
#include<graphics.h>
#include<conio.h>
#define STARTANG 15
#define ENDANG 135

void main(void)
{
   int driver=DETECT, mode;
   int xC=200, yC=200;
   int radius=100;

   initgraph(&driver, &mode,“c:\\tc\\bgi");
   arc(xC, yC, STARTANG, ENDANG, radius);
   getch();
   closegraph();
}
```

Drawing Pixels

- Individual pixels can be plotted using putpixel() function
- Helps to create complex image

```
void far putpixel(int x, int xy, int color)
```

- where

x, y	Coordinates of the pixel
color	color of pixel

- Another function getpixel() is used to find the color of a pixel at a point

Text with Graphics

- The function settextstyle() is used to select different fonts, specify the orientation and size of text.

```
void far settextstyle(int font, int direction,
                                  int charsize)
```

- where

font	which font to use
direction	the orientation of text
charsize	the size of characters used (HORIZ_DIR for left to right and VERT_DIR for bottom to top)

Text with Graphics: Example

```
#include<graphics.h>
#include<conio.h>
void main(void){
   int driver=DETECT, mode, fontsize;
   initgraph(&driver, &mode,“c:\\tc\\bgi");
   fontsize=6;
   settextstyle(GOTHIC_FONT, HORIZ_DIR, fontsize);
   outtext("Gothic ");
   fontsize=6;
   settextstyle(TRIPLEX_FONT, HORIZ_DIR, fontsize);
   outtext("Triplex ");
   moveto(0,70);
   settextstyle(SMALL_FONT, HORIZ_DIR, fontsize);
   outtext("Small ");
   getch();
   closegraph();
}
```