**Các hàm trong chế độ đồ họa**

**\*Hằng biểu diễn màu:**

|  |  |
| --- | --- |
|  |  |

**\*Hàm về màu :**

**setcolor(int color);**

là hàm đặt màu vẽ hiện tại. Màu số 0 là màu nền. Khoảng màu chạy từ 0 ..15

Ngoài ra, còn có 1 số hàm về màu sắc khác:

**getmaxcolor();**

là hàm trả về giá trị màu cao nhất cho thiết bị và chế độ hiện tại. ( theo bảng màu trên getmaxcolor() =15).

**setbkcolor(int color);**

là hàm xác lập màu nền. Khoảng màu chạy từ 0 ..15

**\*Hàm về text:**

**outtextxy(int x, int y, char far \*textstring);**

sẽ hiển thị nội dung của xâu textstring tại tọa độ (x,y). Do đó lệnh outtextxy có thể tương đương với việc thực hiện 2 lệnh: moveto(x,y) và outtext(textstring).

**settextstyle(int font, int direction, int charsize);**

thiết lập kiểu chữ hiện ra trên màn hình.

-Kiểu font chữ:



-Chiều viết (direction):

* HORIZ\_DIR=0 : nằm ngang, từ trái qua phải
* VERT\_DIR=1 : thẳng đứng, từ dưới lên trên

-Kích thước chữ (charsize): nhận giá trị từ 1 đến 10, là hệ số phóng to chữ.

Ngoài ra , còn có 1 số hàm về text khác như:

**outtext(char far \*textstring);**

sẽ hiển thị nội dung của xâu textstring tại vị trí hiện tại của con trỏ đồ họa (thường sẽ ở vị trí (0,0).

**settextjusttify (int Horz, int Vert);**

là hàm xác định vị trí dòng văn bản viết ra ,theo 2 hàm outtext nói ở trên.

-Tham số Horz có thể nhận các giá trị sau:

* LEFT\_TEXT=0 : văn bản xuất hiện ở bên trái con trỏ đồ họa
* CENTER\_TEXT=1: văn bản xuất hiện với tâm là vị trí con trỏ đồ họa
* RIGHT\_TEXT=2: văn bản xuất hiện ở bên phải con trỏ đồ họa

-Còn Vert là tham số có thể nhận các giá trị sau:

* BOTTOM\_TEXT=0: văn bản xuất hiện ở phía trên con trỏ
* CENTER\_TEXT=1: văn bản xuất hiện ở quanh con trỏ
* TOP\_TEXT=2: văn bản xuất hiện ở phía dưới con trỏ

**\*Hàm  vẽ đường thẳng:**

**line(int x1, int y1, int x2, int y2);**

là hàm vẽ đường thẳng nối liền 2 điểm có tọa độ (x1,y1) và (x2,y2). hàm này không phụ thuộc vào vị trí ban đầu của con trỏ màn hình. Sau khi vẽ xong, con trỏ màn hình nằm ở vị trí điểm (x2,y2). Lưu ý là con trỏ đồ họa chỉ được xác lập vị trí hiện tại trên màn hình đồ họa, chứ không có điểm sáng nhấp nháy vật lý như con trỏ trong màn hình văn bản nha.

Ngoài ra còn 1 số hàm vẽ đường thẳng khác như:

**lineto(int x, int y);**

vẽ đường thẳng từ vị trí con trỏ hiện tại tới điểm (x,y).Sau khi vẽ xong, con trỏ màn hình nằm ở vị trí điểm (x,y)

**linerel(int dx, int dy);**

vẽ đường thẳng từ vị trí con trỏ hiện tại tới điểm (x + dx ,y +dy).Sau khi vẽ xong, con trỏ màn hình nằm ở vị trí điểm (x+dx ,y+dy)

**\*Hàm vẽ :**

**circle(int x, int y, int R);** vẽ đường tròn có tâm tại (x,y) bán kính là R

– Hàm **void arc(int x, int y, int gd, int gc, int r)** : vẽ một cung tròn với tâm có toạ độ (x,y) , bán kính r, từ góc đầu tiên là gr đến góc cuối là gc (góc tính bằng độ). Màu của nét vẽ do hàm setcolor() đặt.

– Hàm**ellipse(int x, int y, int gd, int gc, int rx, int ry)**: vẽ một cũng ellipse với tâm là (x,y) từ góc đầu gd đến góc cuối gc, bán kính trục x là rx, bán kính trục y là ry.

– Hàm **void rectangle(int x1, int y1, int x2, int y2) :**vẽ một đường chữ nhật có đỉnh trên bên trái là (x1,y1) đỉnh dưới bên phải là (x2,y2).

**Hàm vẽ miền**

– Hàm **void setfillstyle(int mẫu, int màu)**: đặt mẫu tô và màu tô cho các hình đặc và miền đóng.    
+Màu có giá trị từ 0 đến 15 (xem lại ở bảng màu).   
+ Mẫu có giá trị từ 0 đến 12 được cho ở bảng dưới. Với mẫu có giá trị 12 thì tự thiết kế mẫu tô theo 8 byte.

– Hàm **void pieslice(int x, int y, int gd, int gc, int r)**: vẽ và tô màu một hình quạt có tâm là (x,y), bá kính r, từ góc đầu gd đến góc cuối gc.

– Hàm **void sector(int x, int y, int gd, int gc, int rx, int ry)** : vẽ và tô màu một mảnh ellipse có tâm là (x,y), từ góc đầu đến gd, đến góc cuối gc, có bán kính trục x là rx, bán kính trục y là ry.

– Hàm **fillellipse(int x, int y, int rx, int ry)** : hàm vẽ và tô màu một elip có tâm (x,y) và bán kính hai trục là rx, ry.

– Hàm void bar(int x1, int y1, int x2, int y2) vẽ và tô màu một hình chữ nhật.

– Hàm **void bar3d(int x1, int y1, int x2, int y2, int depth, int top)** : vẽ một khối hộp chữ nhật, màu vẽ xác định bởi hàm setcolor. Mặt trước có toạ độ góc trên trái là (x1,y1) góc dưới phải là (x2,y2), được tô màu bởi hàm setfillstyle. Chiều sâu của khối hộp là depth điểm. Nếu top =0 (TOPOFF) thì hộp không có lắp.

Hàm vẽ đường gấp khúc, tô màu nền

– Hàm **void drawpoly(int n, int a[])**: vẽ đường gấp khúc qua n điểm (x1, y1), (x2,y2), … , (xn,yn). Trong đó a={x1,y1,x2,y2,….xn,yn}. Nếu điểm cuối cùng trùng với điểm đầu ta được đường gấp khúc khép kín.

– Hàm **void fillpoly(int n, int a[])**vẽ và tô màu một đa giác có n đỉnh (x1,y1), ( x2,y2) , … , (xn,yn). Trong đó a={x1,y1,x2,y2,….xn,yn}.

– Hàm **floodfill(int x, int y, int màu\_biên)** : tô màu một miền kín trên màn hình được bao quanh bởi một đường có màu là màu\_biên, (x,y) là một điểm tuỳ ý ở bên trong miền kín. Màu tô và mẫu tô xác định bởi hàm setfillstyle. Nếu (x,y) nằm ngoài miền kín thì vùng ngoài miền kín được tô. Nếu trên màn hình không có miền kín như đã chỉ thì cả màn hình được tô màu.