

```

Joaquín
#include <stdio.h>
#include <graphics.h>
int main()
{
    int gd = DETECT, gm;
    int graph = (b gd, b gm, " ");
    line(0, 200, getmaxx(), 200);
    line(0, 360, getmaxx(), 360);
    setcolor(WHITE);
    rectangle(150, 210, 260, 230);
    floodfill(152, 220, WHITE);
    rectangle(150, 240, 260, 260);
    floodfill(152, 241, WHITE);
    rectangle(150, 270, 260, 290);
    floodfill(152, 271, WHITE);
    rectangle(150, 300, 260, 320);
    floodfill(152, 301, WHITE);
    rectangle(150, 330, 260, 350);
    floodfill(152, 331, WHITE);
    setcolor(WHITE);
    rectangle(140, 200, 145, 130);
    rectangle(130, 130, 155, 70);
    setcolor(RED);
    circle(142, 82, 6);
    floodfill(142, 82, RED);
    setcolor(YELLOW);
    circle(142, 100, 6);
    floodfill(142, 100, YELLOW);
}

```

```
setcolor (GREEN);  
circle (142, 118, 6);  
fill (143, 118, GREEN);  
setcolor (WHITE);  
getch();  
closegraph();  
return 0;  
}
```



Algorithm DDA

Step 1 Start

Step 2 Declare $x_1, y_1, x_2, y_2, dx, dy, x, y$ as integer variables

Step 3 Enter value of x_1, y_1, x_2, y_2

Step 4 Calculate $dx = x_2 - x_1$

Step 5 Calculate $dy = y_2 - y_1$

Step 6 : If $(dx) > dy$
Step 7 = (dx)
else

Step 7 : $x_{inc} = dx / \text{Step}$
 $y_{inc} = dy / \text{Step}$
assign $x = x_1$
assign $y = y_1$

Step 8 : Set $\text{Pixel}(x, y)$

Step 9 : $x = x + x_{inc}$
 $y = y + y_{inc}$
Set $\text{Pixel}(\text{Round}(x), \text{Round}(y))$

Step 10 : Repeat step 9 until $x = x_2$

Step 11 : End

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

DDA

```
int main()
```

```
{
    int rou(float num)
    {
        return num < 0 ? num - 0.5 : num + 0.5;
    }
    int x1 = 100, x2 = 250, y1 = 100, y2 = 250, step;
    int gd = DETECT, gm;
    float x, y, m;
    int dx = x2 - x1;
    int dy = y2 - y1;
    m = dy / dx;
    if (dx > dy)
        step = dx;
    else
        step = dy;
    int i;
    graph(gd, gm, "");
    outtextxy(x1, y1, "A");
    outtextxy(x2, y2, "B");
    putpixel(x1, y1, RED);
    x = x1, y = y1;
    while (step > 0)
    {
        if (m < 1)
        {
            x = x + 1;
            y = y + m;
        }
        else if (m >= 1)
        {
            x = x + 1/m;
            y = y + 1;
        }
        putpixel(rou(x), rou(y), RED);
        step--;
    }
    getch();
    return 0;
}
```

