Algorithm for flood fill function:

Step1 : ~~delare~~ start

Step2 : declare variable current

Step3 : call the predefined function getpixel(x, y) and store in variable current.

Step4 : if current == old

     ₹. delay by 5

     ○ put the pixel

putpixel(x, y, newcol);
floodfill(x+1, y, old, newcol);
floodfill(x-1, y, old, newcol);
floodfill(x, y+1, old, newcol);
floodfill(x, y-1, old, newcol);
floodfill(x+1, y+1, old, newcol);
floodfill(x-1, y+1, old, newcol);
floodfill(x+1, y-1, old, newcol);
floodfill(x-1, y-1, old, newcol);
}

Step5 : stop

Algorithm for program

Step 1 : Start

step 2 : define the flood fill function

step 3 : declared variable gd, gm

step 4 : draw the rectangle with predefine function.

Step 5 : call the flood fill function with
          required cordinate

step 6 : stop

Neeraj pant

Program :

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

void floodfill (int x, int y, int old, int newcol)
{
    int current;
    current = getpixel (x, y);
    if (current == old)
    {
        delay (5)
        putpixel (x, y, newcol);
        floodfill (x+1, y, old, newcol);
        floodfill (x-1, y, old, newcol);
        floodfill (x, y+1, old, newcol);
        floodfill (x, y-1, old, newcol);
        floodfill (x+1, y+1, old, newcol);
        floodfill (x-1, y+1, old, newcol);
        floodfill (x+1, y-1, old, newcol);
        floodfill (x-1, y-1, old, newcol);
    }
void main ()
    {
        int gd = DETECT, gm;
```

Neeraj Pant

```
initgraph(&gd, &gm, " ");
rectangle(50, 50, 150, 150);
floodfill(70, 70, 0, 15)
getch()
closegraph();
}
```

Neeraj Pant