Name: Khush Nitinkumar Patel

PRN: 2020BTECS00037

Batch: T5

Experiment 9:Implement Curves

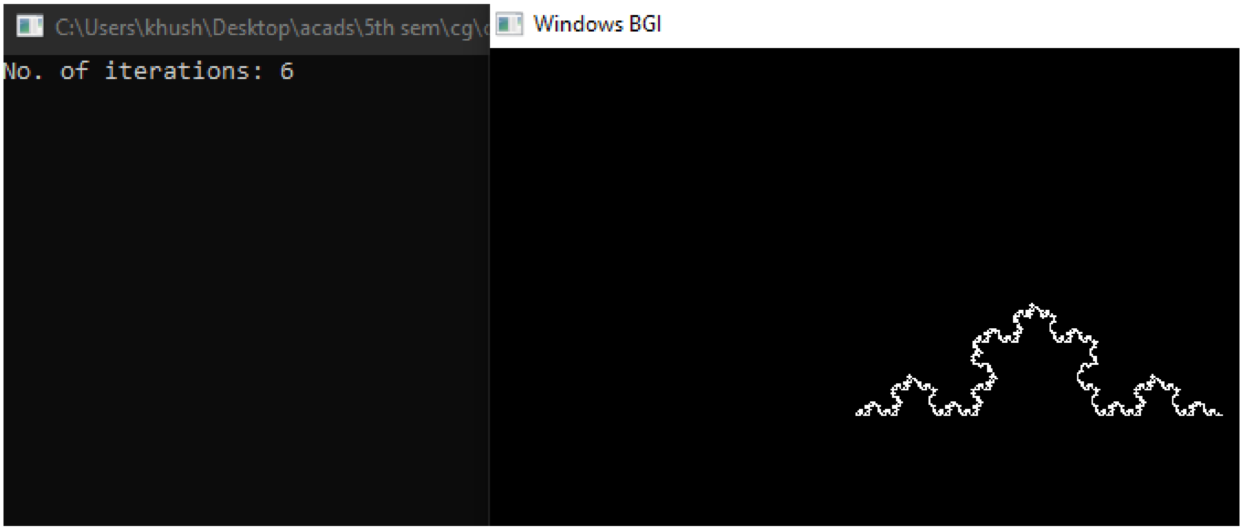
- ### Koch Curve

  Code

```
#include <stdio.h>
#include <stdlib.h>
#include <dos.h>
#include <graphics.h>
#include <conio.h>
#include <math.h>

void koch(int x1, int y1, int x2, int y2, int it)
{
float angle = 60 * M_PI / 180;
int x3 = (2 * x1 + x2) / 3;
int y3 = (2 * y1 + y2) / 3;

int x4 = (x1 + 2 * x2) / 3;
int y4 = (y1 + 2 * y2) / 3;

int x = x3 + (x4 - x3) * cos(angle) + (y4 - y3) * sin(angle); int y = y3 - (x4 - x3) * sin(angle) + (y4 - y3)
* cos(angle);

if (it > 0)
{
koch(x1, y1, x3, y3, it - 1);
koch(x3, y3, x, y, it - 1);
koch(x, y, x4, y4, it - 1);
koch(x4, y4, x2, y2, it - 1);
}
else
{
```

```
line(x1, y1, x3, y3);
line(x3, y3, x, y);
line(x, y, x4, y4);
line(x4, y4, x2, y2);
}
}

int main(void)
{
int gd = DETECT, gm;
initgraph(&gd, &gm, "c:\\turboc3\\bgi");
int x1 = 200, y1 = 200, x2 = 400, y2 = 200, n;
printf("No. of iterations: "); scanf("%d", &n);
koch(x1, y1, x2, y2, n);

getch(); closegraph(); return 0;
}
```

Output



```
No. of iterations: 6
```

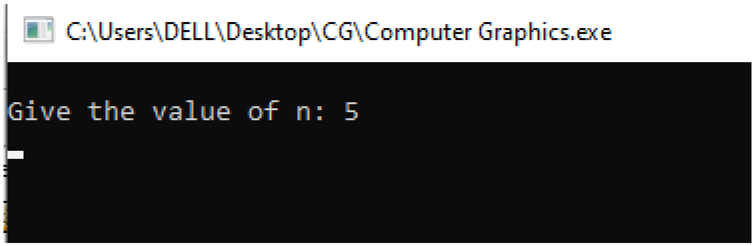
Hilbert Curve

Code

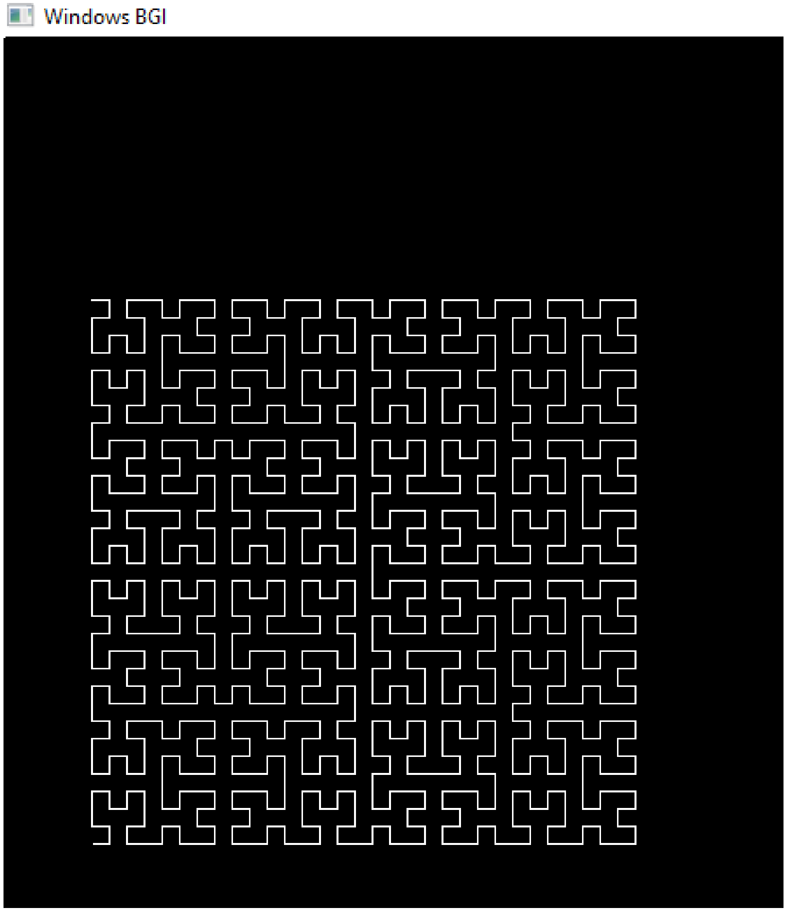
```
#include <stdio.h>
#include <stdlib.h>
#include <dos.h>
#include <graphics.h> #include <conio.h>

void move(int a, int b, int &x, int &y)
{
if (a == 1) y -= b;
else if (a == 2) x += b;
else if (a == 3) y += h;
else if (a == 4) x -= b;
lineto(x, y); delay(10);
}

void hilbert(int r, int d, int l, int u, int i, int h, int &x, int &y)
{
if (i > 0)
{
i--;
hilbert(d, r, u, l, i, h, x, y);
move(r, h, x, y);
hilbert(r, d, l, u, i, h, x, y);
move(d, h, x, y); hilbert(r, d, l, u, i, h, x, y);
move(l, h, x, y);
hilbert(u, l, d, r, i, h, x, y);
}
}
```

```
int main()
{
int gd = DETECT, gm;
initgraph(&gd, &gm, "c:\\turboc3\\bgi");
int n, x1, y1;
int x0 = 50, y0 = 150, x, y, h = 10, r = 2, d = 3, l = 4, u = 1;
printf("\nGive the value of n: ");
scanf("%d", &n);
x = x0; y = y0;
moveto(x, y);
hilbert(r, d, l, u, n, h, x, y);
getch();
closegraph()
```

Outpu



```

Give the value of n: 5
```



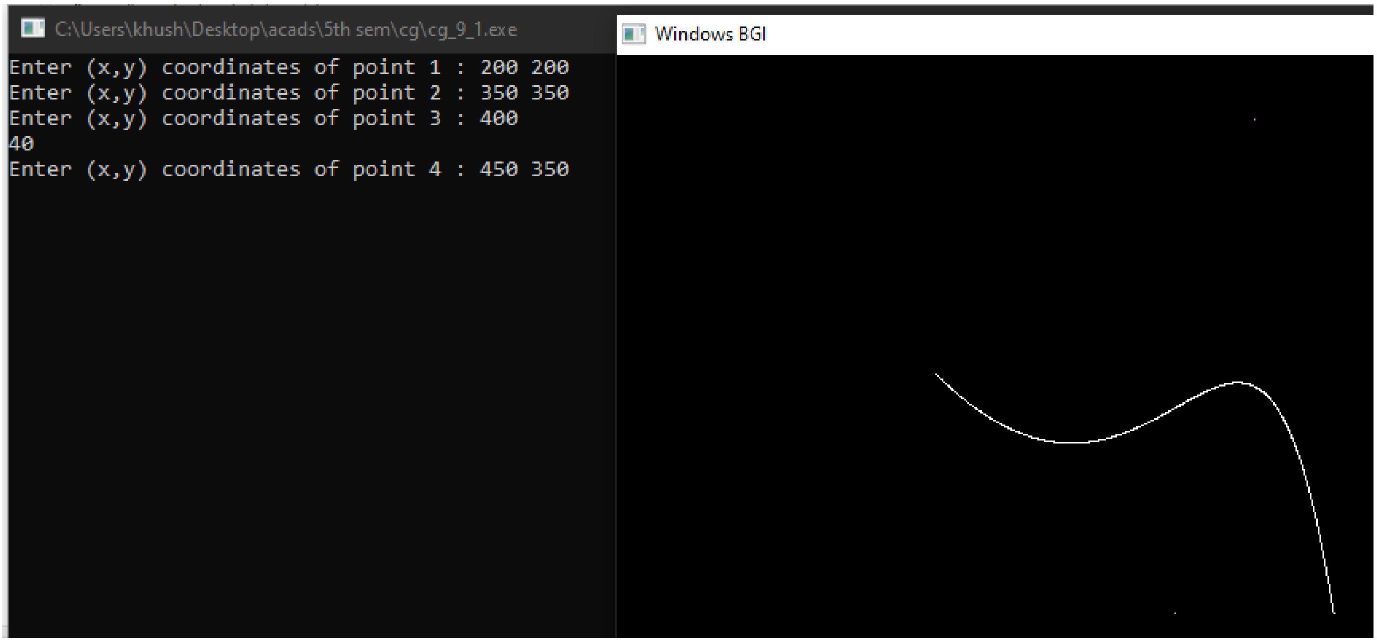
- **Bezier Curve**

  Code:

```
#include <stdio.h>
#include <graphics.h>
#include <conio.h>
int maxx, maxy;
float xxx[4][2];

void line1(float x2, float y2)
{
   line(xxx[0][0], xxx[0][1], x2, y2);
   delay(0.5);
   xxx[0][0] = x2;
   xxx[0][1] = y2;
}

void bezier(float xb, float yb, float xc, float yc, float xd, float yd, int n)
{
   float xab, yab, xbc, ybc, xcd, ycd;
   float xabc, yabc, xbcd, ybcd;
   float xabcd, yabcd;
   if (n == 0)
   {
      line1(xb, yb);
      line1(xc, yc);
      line1(xd, yd);
   }
   else
   {
      xab = (xxx[0][0] + xb) / 2;
      yab = (xxx[0][1] + yb) / 2;
      xbc = (xb + xc) / 2;
      ybc = (yb + yc) / 2;
      xcd = (xc + xd) / 2;
      ycd = (yc + yd) / 2;
      xabc = (xab + xbc) / 2;
      yabc = (yab + ybc) / 2;
      xbcd = (xbc + xcd) / 2;
      ybcd = (ybc + ycd) / 2;
      xabcd = (xabc + xbcd) / 2;
      yabcd = (yabc + ybcd) / 2;
      n = n - 1;
```

```
        bezier(xab, yab, xabc, yabc, xabcd, yabcd, n);
        bezier(xbcd, ybcd, xcd, ycd, xd, yd, n);
    }
}
int main()
{
    int gd = DETECT, gm;
    initgraph(&gd, &gm, "c:\\turboc3\\bgi");

     float temp1, temp2;
    for (int i = 0; i < 4; i++)
    {
        printf("Enter (x,y) coordinates of point %d : ", i + 1);
        scanf("%f%f", &temp1, &temp2);
        putpixel(temp1, temp2, 13);
        xxx[i][0] = temp1;
        xxx[i][1] = temp2;
    }
    bezier(xxx[1][0], xxx[1][1], xxx[2][0], xxx[2][1], xxx[3][0], xxx[3][1], 8);
    getch();
    closegraph();
}
```

Output



```
Enter (x,y) coordinates of point 1 : 200 200
Enter (x,y) coordinates of point 2 : 350 350
Enter (x,y) coordinates of point 3 : 400
40
Enter (x,y) coordinates of point 4 : 450 350
```