Computer Graphics

Name:- PRANJAL TRIPATHI

Sap ID:- 500096030

Roll no. :- R2142210964

Branch:- B.TECH. CSE

Batch:- BAO-1

Experiment No.6 : Implementation of Bresenham's circle drawing algorithm

```
#include <stdio.h>
#include <dos.h>
#include <graphics.h>
void plotPoints(int cx, int cy, int x, int y) {
    putpixel(cx+x, cy+y, RED);
    putpixel(cx-x, cy+y, RED);
    putpixel(cx+x, cy-y, RED);
    putpixel(cx-x, cy-y, RED);
    putpixel(cx+y, cy+x, RED);
    putpixel(cx-y, cy+x, RED);
```

```
        putpixel(cx+y, cy-x, RED);
        putpixel(cx-y, cy-x, RED);
}
main() {
        int cx, cy, x = 0, y, r, p;
        int gd = DETECT, gm;

        printf("Enter the coordinates of centre of the circle: ");
        scanf("%d %d", &cx, &cy);
        printf("Enter radius of : ");
        scanf("%d", &r);
        y = r;
        p = 3 - 2 * r;
        initgraph(&gd, &gm, "");
        cleardevice();
        while (x < y) {
                plotPoints(cx, cy, x, y);
                x++;
                if (p < 0)
                                p = p + 4 * x + 6; else {
                        y--;
                        p = p + 4 * (x - y) + 10;
                }
                plotPoints(cx, cy, x, y);
                delay(200);
        }getch();
}
```

```c
#include <stdio.h>
#include <dos.h>
#include <graphics.h>
void plotPoints(int cx, int cy, int x, int y) {
        putpixel(cx+x, cy+y, RED);
        putpixel(cx-x, cy+y, RED);
        putpixel(cx+x, cy-y, RED);
        putpixel(cx-x, cy-y, RED);
        putpixel(cx+y, cy+x, RED);
        putpixel(cx-y, cy+x, RED);
        putpixel(cx+y, cy-x, RED);
        putpixel(cx-y, cy-x, RED);
}
main() {
        int cx, cy, x = 0, y, r, p;
        int gd = DETECT, gm;

        printf("Enter the coordinates of centre of the circle: ");
        scanf("%d %d", &cx, &cy);
        printf("Enter radius of : ");
        scanf("%d", &r);
        y = r;
        p = 3 - 2 * r;
        initgraph(&gd, &gm, "");
        cleardevice();
        while (x < y) {
                plotPoints(cx, cy, x, y);
                x++;
                if (p < 0)
                                                p = p + 4 * x + 6; else {
                        y--;
                        p = p + 4 * (x - y) + 10;
                }
                plotPoints(cx, cy, x, y);
                delay(200);
        }getch();
}
```