Nama : **Dino Apriyanto**
NPM : **2257051011**
Kelas : **D**
Mata Kuliah : **Grafikom**

Tugas 3 Algoritma Pembentukan Lingkaran Bresenham dan Midpoint

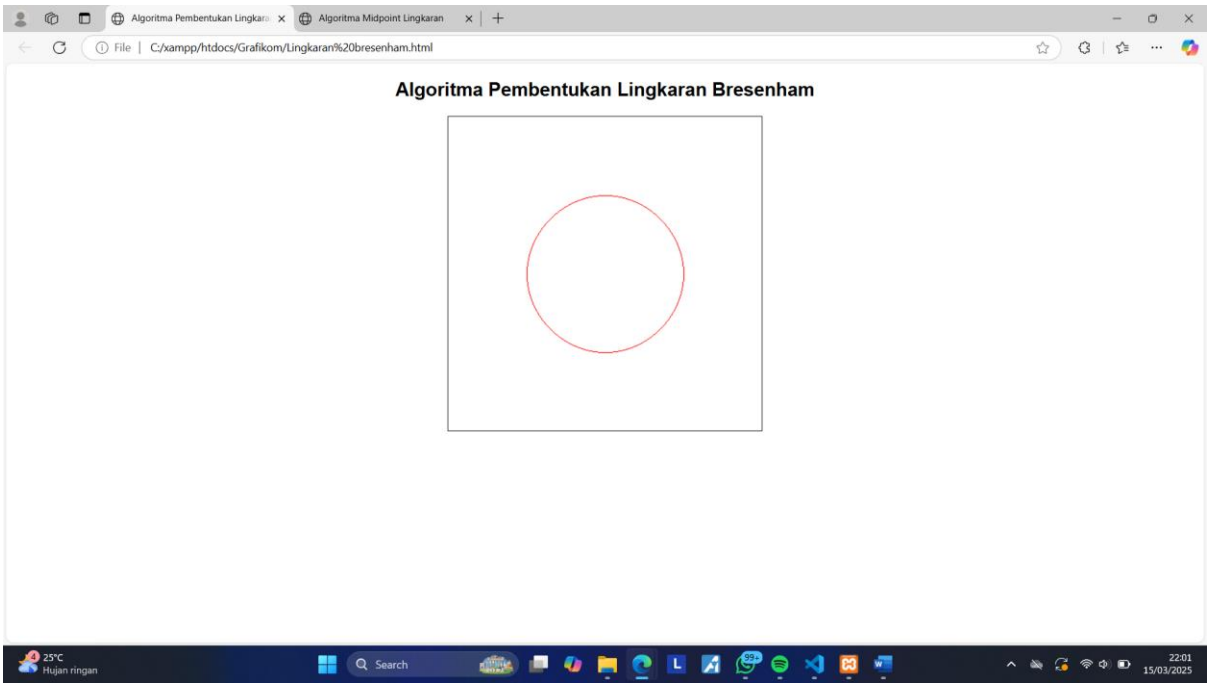
1. Algortima Bresenham

- Source Code :

```html
<!DOCTYPE html>
<html lang="id">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Algoritma Pembentukan Lingkaran Bresenham</title>
    <style>
        body {
            text-align: center;
            font-family: Arial, sans-serif;
        }
        canvas {
            border: 1px solid black;
            background-color: white;
        }
    </style>
</head>
<body>
    <h2>Algoritma Pembentukan Lingkaran Bresenham</h2>
    <canvas id="canvas" width="400" height="400"></canvas>
    <script>
        function drawCircleBresenham(x0, y0, r) {
            let canvas = document.getElementById("canvas");
            let ctx = canvas.getContext("2d");
            let imageData = ctx.getImageData(0, 0, canvas.width, canvas.height);
            let pixels = imageData.data;

            function putPixel(x, y, color) {
                if (x >= 0 && x < canvas.width && y >= 0 && y < canvas.height) {
                    let index = (y * canvas.width + x) * 4;
                    pixels[index] = color[0]; // Red
                    pixels[index + 1] = color[1]; // Green
                    pixels[index + 2] = color[2]; // Blue
                    pixels[index + 3] = 255; // Alpha
                }
            }
```

```
            function plotCirclePoints(x0, y0, x, y) {
                let color = [255, 0, 0]; // Warna merah
                putPixel(x0 + x, y0 + y, color);
                putPixel(x0 - x, y0 + y, color);
                putPixel(x0 + x, y0 - y, color);
                putPixel(x0 - x, y0 - y, color);
                putPixel(x0 + y, y0 + x, color);
                putPixel(x0 - y, y0 + x, color);
                putPixel(x0 + y, y0 - x, color);
                putPixel(x0 - y, y0 - x, color);
            }

            let x = 0;
            let y = r;
            let d = 3 - 2 * r;
            plotCirclePoints(x0, y0, x, y);
            while (y >= x) {
                x++;
                if (d > 0) {
                    y--;
                    d = d + 4 * (x - y) + 10;
                } else {
                    d = d + 4 * x + 6;
                }
                plotCirclePoints(x0, y0, x, y);
            }

            ctx.putImageData(imageData, 0, 0);
        }

        drawCircleBresenham(200, 200, 100);
    </script>
</body>
</html>
```

- Hasil

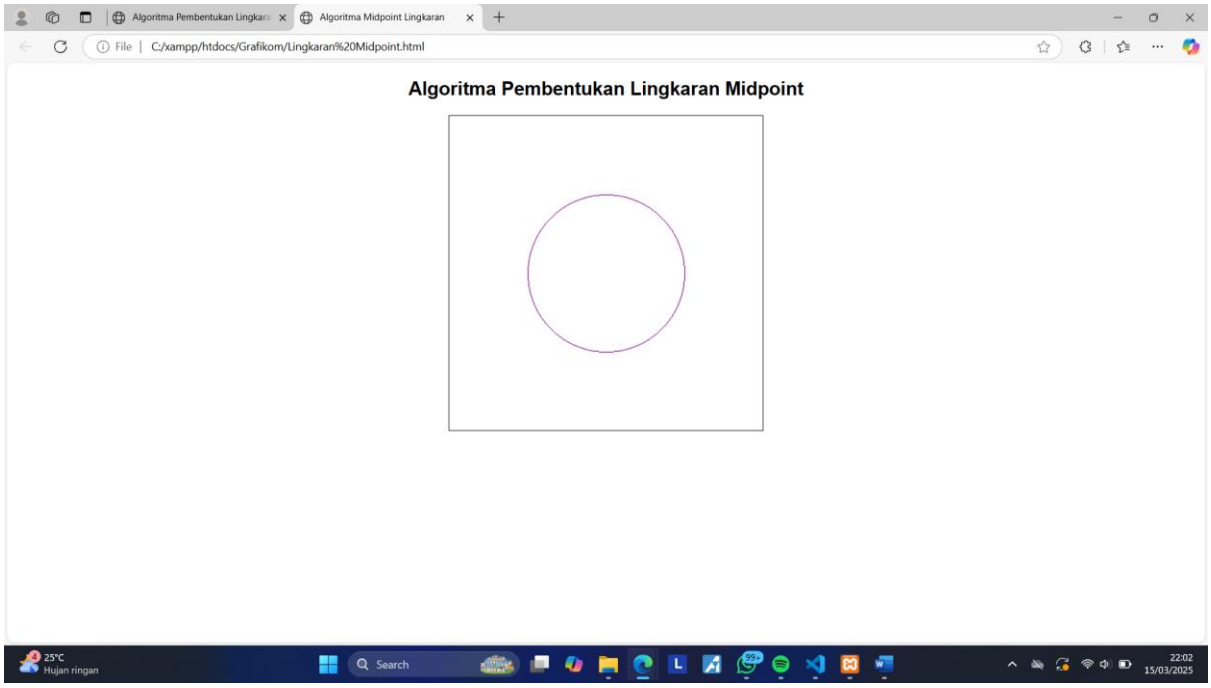


2. Algoritma Midpoint

- Source Code :

```html
<!DOCTYPE html>
<html lang="id">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Algoritma Midpoint Lingkaran</title>
    <style>
        body {
            text-align: center;
            font-family: Arial, sans-serif;
        }
        canvas {
            border: 1px solid black;
            background-color: white;
        }
    </style>
</head>
<body>
    <h2>Algoritma Pembentukan Lingkaran Midpoint</h2>
    <canvas id="canvas" width="400" height="400"></canvas>
    <script>
        function drawCircleMidpoint(x0, y0, r) {
            let canvas = document.getElementById("canvas");
            let ctx = canvas.getContext("2d");

            function putPixel(x, y) {
                ctx.fillStyle = "purple";
                ctx.fillRect(x, y, 1, 1);
            }

            function plotCirclePoints(x0, y0, x, y) {
                putPixel(x0 + x, y0 + y);
                putPixel(x0 - x, y0 + y);
                putPixel(x0 + x, y0 - y);
                putPixel(x0 - x, y0 - y);
                putPixel(x0 + y, y0 + x);
                putPixel(x0 - y, y0 + x);
                putPixel(x0 + y, y0 - x);
                putPixel(x0 - y, y0 - x);
            }

            let x = 0;
            let y = r;
            let p = 1 - r;
            plotCirclePoints(x0, y0, x, y);

            while (x < y) {
                x++;
                if (p < 0) {
                    p += 2 * x + 1;
                } else {
                    y--;
                    p += 2 * (x - y) + 1;
                }
                plotCirclePoints(x0, y0, x, y);
            }
        }

        drawCircleMidpoint(200, 200, 100);
    </script>
</body>
</html>
```

- Hasil :


Algoritma Pembentukan Lingkaran
Algoritma Midpoint Lingkaran
File | C:/xampp/htdocs/Grafikom/Lingkaran%20Midpoint.html
Algoritma Pembentukan Lingkaran Midpoint
25°C
Hujan ringan
Search
22:02
15/03/2025