Nama : Muhammad Hafiz Zidane

NPM : 2217051036

Kelas : CD

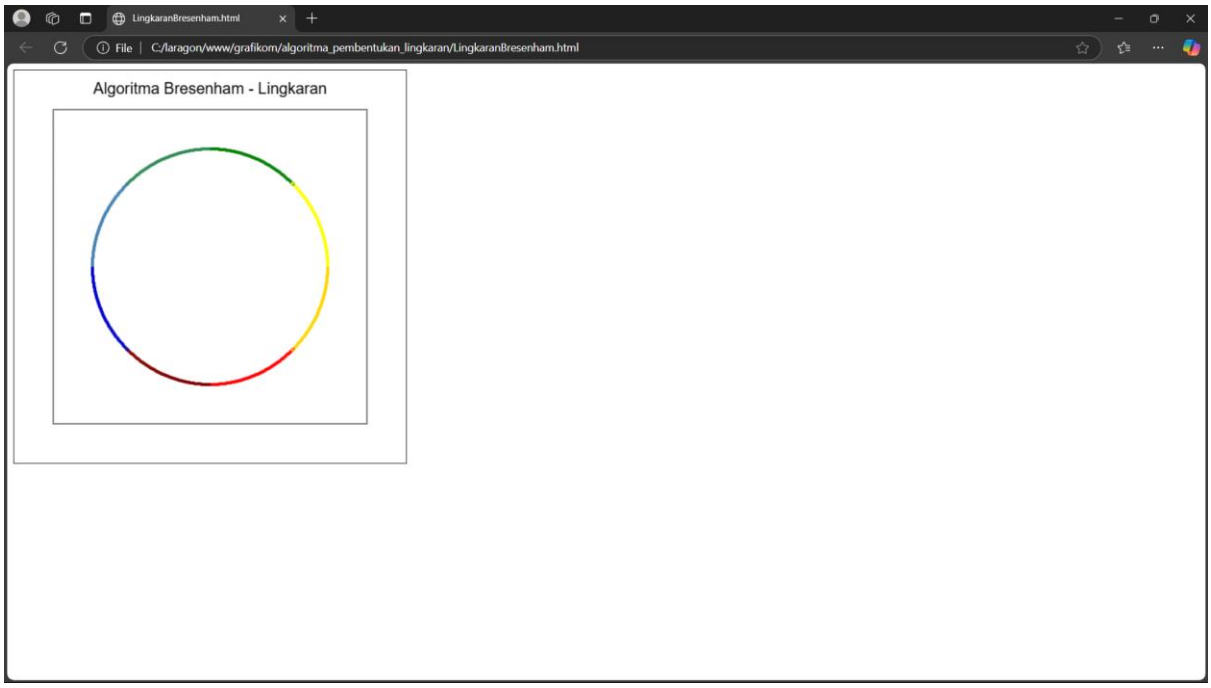
Tugas 3 Algoritma Pembentukan Lingkaran

1. **Algoritma Bresenham**

```
<!DOCTYPE html>
<html>
<body>
    <canvas id="myCanvas" width="500" height="500" style="border:1px solid
black; display:block;"></canvas>
    <script>
        var canvas = document.getElementById("myCanvas");
        var context = canvas.getContext("2d");

        context.strokeStyle = "black";
        context.strokeRect(50, 50, 400, 400);

        context.font = "20px Arial";
        context.fillStyle = "black";
        context.textAlign = "center";
        context.fillText("Algoritma Bresenham - Lingkaran", canvas.width /
2, 30);

        function putPixel(context, x, y, color, size = 3) {
            context.fillStyle = color;
            context.fillRect(x - size / 2, y - size / 2, size, size);
        }

        function circlePlotPoints(context, x0, y0, x, y) {
            let size = 3;
            putPixel(context, x0 + x, y0 + y, "red", size);
            putPixel(context, x0 - x, y0 + y, "maroon", size);
            putPixel(context, x0 + x, y0 - y, "green", size);
            putPixel(context, x0 - x, y0 - y, "SeaGreen", size);
            putPixel(context, x0 + y, y0 + x, "gold", size);
            putPixel(context, x0 - y, y0 + x, "MediumBlue", size);
            putPixel(context, x0 + y, y0 - x, "yellow", size);
            putPixel(context, x0 - y, y0 - x, "SteelBlue", size);
        }

        function circleBres(context, x0, y0, r) {
            let x = 0;
            let y = r;
            let d = 3 - 2 * r;
            circlePlotPoints(context, x0, y0, x, y);
```

```
            while (y >= x) {
                x++;
                if (d > 0) {
                    y--;
                    d = d + 4 * (x - y) + 10;
                } else {
                    d = d + 4 * x + 6;
                }
                circlePlotPoints(context, x0, y0, x, y);
            }
        }

        circleBres(context, 250, 250, 150);
    </script>
</body>
</html>
```

Output:

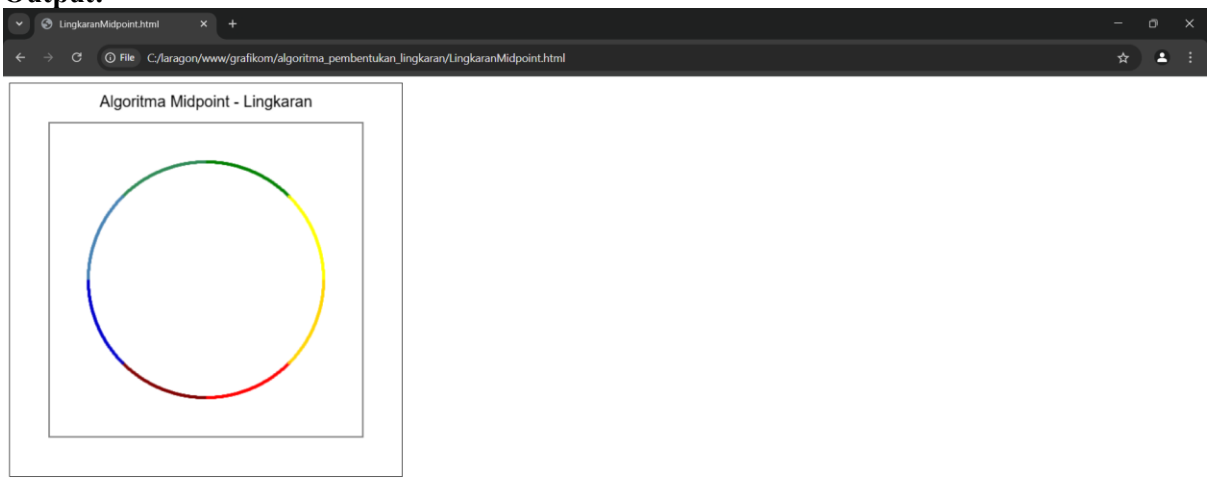


2. Algoritma Midpoint

```
<!DOCTYPE html>
<html>
<body>
    <canvas id="myCanvas" width="500" height="500" style="border:1px solid
black; display:block;"></canvas>
    <script>
        var canvas = document.getElementById("myCanvas");
        var context = canvas.getContext("2d");
```

```
        context.strokeStyle = "black";
        context.strokeRect(50, 50, 400, 400);

        context.font = "20px Arial";
        context.fillStyle = "black";
        context.textAlign = "center";
        context.fillText("Algoritma Midpoint - Lingkaran", canvas.width /
2, 30);

        function putPixel(context, x, y, color, size = 3) {
            context.fillStyle = color;
            context.fillRect(x - size / 2, y - size / 2, size, size);
        }

        function circlePlotPoints(context, x0, y0, x, y) {
            let size = 3;
            putPixel(context, x0 + x, y0 + y, "red", size);
            putPixel(context, x0 - x, y0 + y, "maroon", size);
            putPixel(context, x0 + x, y0 - y, "green", size);
            putPixel(context, x0 - x, y0 - y, "SeaGreen", size);
            putPixel(context, x0 + y, y0 + x, "gold", size);
            putPixel(context, x0 - y, y0 + x, "MediumBlue", size);
            putPixel(context, x0 + y, y0 - x, "yellow", size);
            putPixel(context, x0 - y, y0 - x, "SteelBlue", size);
        }

        function circleMidpoint(context, x0, y0, radius) {
            let x = 0;
            let y = radius;
            let p = 1 - radius;
            circlePlotPoints(context, x0, y0, x, y);

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

        circleMidpoint(context, 250, 250, 150);
    </script>
</body>
</html>
```

Output:


LingkaranMidpoint.html
File
C:/laragon/www/grafikom/algoritma_pembentukan_lingkaran/LingkaranMidpoint.html
Algoritma Midpoint - Lingkaran