Tugas 2 Grafika Komputer
Algoritma Pembentukan Garis

Nama : Angga Bagus Susilo

Kelas : CD

NPM : 2217051064

1. Bresenham Algorithm

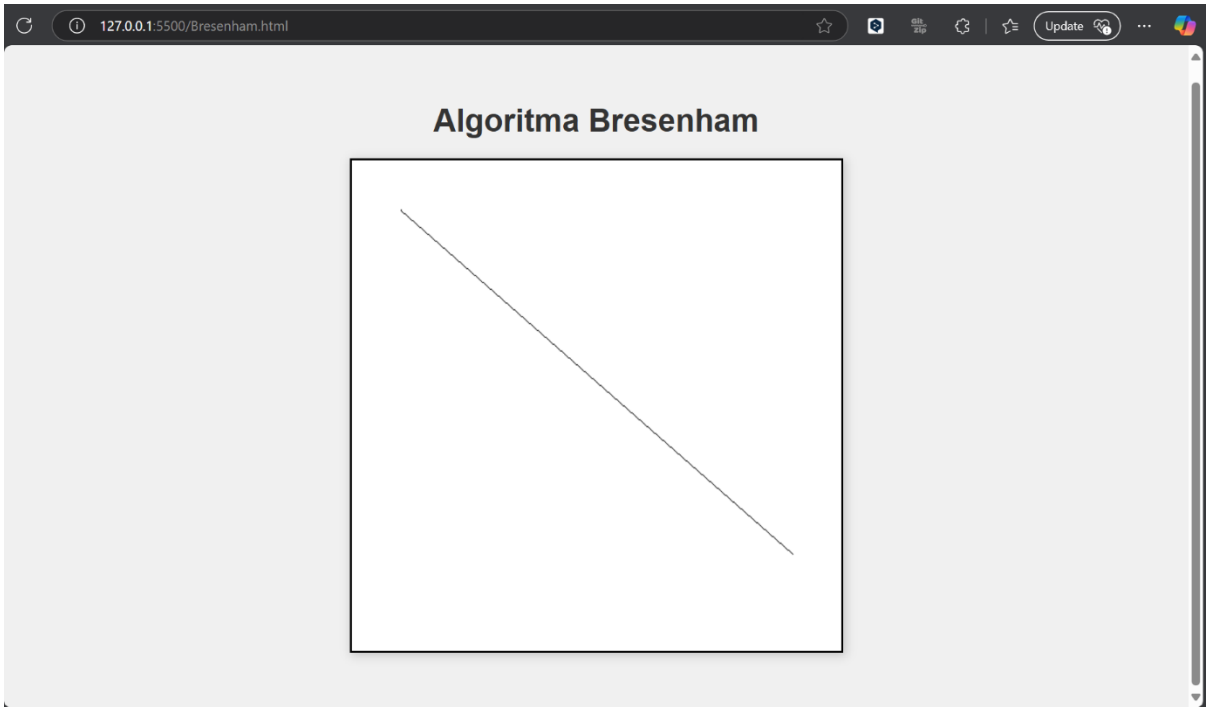
a) Source Code

Bresenham.html DDA.html

Bresenham.html > html > body > script > drawLineBresenham

```html
<!DOCTYPE html>
<html lang="id">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Algoritma Bresenham</title>
    <style>
        body {
            display: flex;
            justify-content: center;
            align-items: center;
            height: 100vh;
            background-color: #f0f0f0;
            font-family: Arial, sans-serif;
        }
        .container {
            text-align: center;
        }
        canvas {
            border: 2px solid black;
            background-color: white;
            box-shadow: 2px 2px 10px rgba(0, 0, 0, 0.2);
        }
        h1 {
            margin-bottom: 20px;
            color: #333;
        }
    </style>
</head>
<body>
    <div class="container">
        <h1>Algoritma Bresenham</h1>
        <canvas id="canvas" width="500" height="500"></canvas>
    </div>
    <script>
        function drawPixel(ctx, x, y) {
            ctx.fillRect(x, y, 1, 1);
        }

        function drawLineBresenham(x0, y0, x1, y1) {
            const canvas = document.getElementById("canvas");
            const ctx = canvas.getContext("2d");
            ctx.fillStyle = "black";

            let x = x0;
            let y = y0;
            drawPixel(ctx, x, y);

            let dx = x1 - x0;
            let dy = y1 - y0;
            let D = 2 * dy - dx;

            for (let i = x0; i <= x1; i++) {
                if (D >= 0) {
                    y += 1;
                    D = D + (2 * dy - 2 * dx);
                } else {
                    D = D + 2 * dy;
                }
                drawPixel(ctx, i, y);
            }
        }

        // Contoh penggunaan
        drawLineBresenham(50, 50, 450, 400);
    </script>
</body>
</html>
```

b) Output



2. DDA Algorithm

a) Source Code

Bresenham.html | DDA.html

DDA.html > ...

```html
<!DOCTYPE html>
<html lang="id">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Algoritma DDA</title>
    <style>
        body {
            display: flex;
            justify-content: center;
            align-items: center;
            height: 100vh;
            background-color: #f0f0f0;
            font-family: Arial, sans-serif;
        }
        .container {
            text-align: center;
        }
        canvas {
            border: 2px solid black;
            background-color: white;
            box-shadow: 2px 2px 10px rgba(0, 0, 0, 0.2);
        }
        h1 {
            margin-bottom: 20px;
            color: #333;
        }
    </style>
</head>
<body>
    <div class="container">
        <h1>Algoritma DDA</h1>
        <canvas id="canvas" width="500" height="500"></canvas>
    </div>
    <script>
        function drawPixel(ctx, x, y) {
            ctx.fillRect(Math.round(x), Math.round(y), 1, 1);
        }

        function drawLineDDA(x1, y1, x2, y2) {
            const canvas = document.getElementById("canvas");
            const ctx = canvas.getContext("2d");
            ctx.fillStyle = "black";

            let dx = x2 - x1;
            let dy = y2 - y1;
            let steps = Math.max(Math.abs(dx), Math.abs(dy));
            let xIncrement = dx / steps;
            let yIncrement = dy / steps;

            let x = x1;
            let y = y1;
            for (let i = 0; i <= steps; i++) {
                drawPixel(ctx, x, y);
                x += xIncrement;
                y += yIncrement;
            }
        }

        // Contoh penggunaan
        drawLineDDA(450, 50, 50, 400);
    </script>
</body>
</html>
```

b) Output