

Nama : Ngasiroh Nurjayatri
Npm : 2257051032
Kelas : D
Tugas : Algoritma Pembentukan Lingkaran

1. Algoritma Bresenham

- Source Code

```

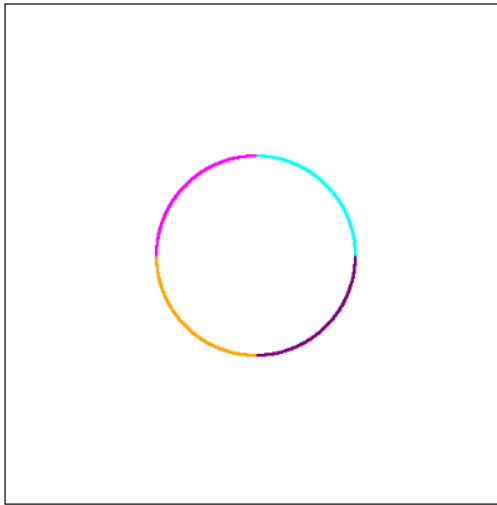
bresenham3.html x midpoint.html
bresenham3.html > html > body > script > plotCirclePoints
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4    <meta charset="UTF-8">
5    <meta name="viewport" content="width=device-width, initial-scale=1.0">
6    <title>Bresenham Circle Algorithm</title>
7  </head>
8  <body>
9    <h2>Algoritma Bresenham</h2>
10   <canvas id="bresenhamCanvas" width="500" height="500" style="border: 1px solid black;"></canvas>
11   <script>
12     const canvas = document.getElementById("bresenhamCanvas");
13     const ctx = canvas.getContext("2d");
14
15     function drawPixel(x, y, color = "black") {
16       ctx.fillStyle = color;
17       ctx.fillRect(x, y, 3, 3); // Menggunakan ukuran pixel yang lebih besar
18     }
19
20     function plotCirclePoints(xc, yc, x, y) {
21       let colors = ["purple", "orange", "cyan", "magenta"];
22       drawPixel(xc + x, yc + y, colors[0]);
23       drawPixel(xc - x, yc + y, colors[1]);
24       drawPixel(xc + x, yc - y, colors[2]);
25       drawPixel(xc - x, yc - y, colors[3]);
26       drawPixel(xc + y, yc + x, colors[0]);
27       drawPixel(xc - y, yc + x, colors[1]);
28       drawPixel(xc + y, yc - x, colors[2]);
29       drawPixel(xc - y, yc - x, colors[3]);
30     }
31
32     function drawCircleBresenham(xc, yc, r) {
```

```

32     function drawCircleBresenham(xc, yc, r) {
33       let x = 0, y = r;
34       let d = 1 - r; // Menggunakan nilai d yang berbeda
35       plotCirclePoints(xc, yc, x, y);
36
37       while (x < y) {
38         x++;
39         if (d < 0) {
40           d = d + 2 * x + 1;
41         } else {
42           y--;
43           d = d + 2 * (x - y) + 1;
44         }
45         plotCirclePoints(xc, yc, x, y);
46       }
47     }
48
49     drawCircleBresenham(canvas.width / 2, canvas.height / 2, 100);
50   </script>
51 </body>
52 </html>
53
```

- Hasil Pembentukan Lingkaran

Algoritma Bresenham



2. Algoritma Midpoint

- Source Code

```

midpoint.html x
midpoint.html > html > body > canvas#canvasMidpoint
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Midpoint Circle Algorithm</title>
7      <style>
8          canvas { border: 1px solid black; }
9      </style>
10 </head>
11 <body>
12     <h2>Algoritma Midpoint</h2>
13     <canvas id="canvasMidpoint" width="500" height="500"></canvas>
14     <script>
15         const canvasMid = document.getElementById("canvasMidpoint");
16         const ctxMid = canvasMid.getContext("2d");
17
18         function drawPixelMid(x, y, color = "red") {
19             ctxMid.fillStyle = color;
20             ctxMid.fillRect(x, y, 3, 3);
21         }
22
23         function plotMidCircle(xc, yc, x, y) {
24             let colors = ["pink", "purple", "green", "yellow"];
25             drawPixelMid(xc + x, yc + y, colors[0]);
26             drawPixelMid(xc - x, yc + y, colors[1]);
27             drawPixelMid(xc + x, yc - y, colors[2]);
28             drawPixelMid(xc - x, yc - y, colors[3]);
29             drawPixelMid(xc + y, yc + x, colors[0]);
30             drawPixelMid(xc - y, yc + x, colors[1]);
31             drawPixelMid(xc + y, yc - x, colors[2]);
32             drawPixelMid(xc - y, yc - x, colors[3]);

```

```

30     drawPixelMid(xc - y, yc + x, colors[1]);
31     drawPixelMid(xc + y, yc - x, colors[2]);
32     drawPixelMid(xc - y, yc - x, colors[3]);
33 }
34
35 function drawCircleMidpoint(xc, yc, r) {
36     let x = 0, y = r;
37     let p = 1 - r;
38     plotMidCircle(xc, yc, x, y);
39
40     while (x < y) {
41         x++;
42         if (p < 0) {
43             p += 2 * x + 1;
44         } else {
45             y--;
46             p += 2 * (x - y) + 1;
47         }
48         plotMidCircle(xc, yc, x, y);
49     }
50 }
51
52 drawCircleMidpoint(canvasMid.width / 2, canvasMid.height / 2, 100);
53 </script>
54 </body>
55 </html>
56

```

Ln 13, Col 14 Spaces: 4 UTF-8 CRLF H

- Hasil Pembentukan Lingkaran

Algoritma Midpoint

