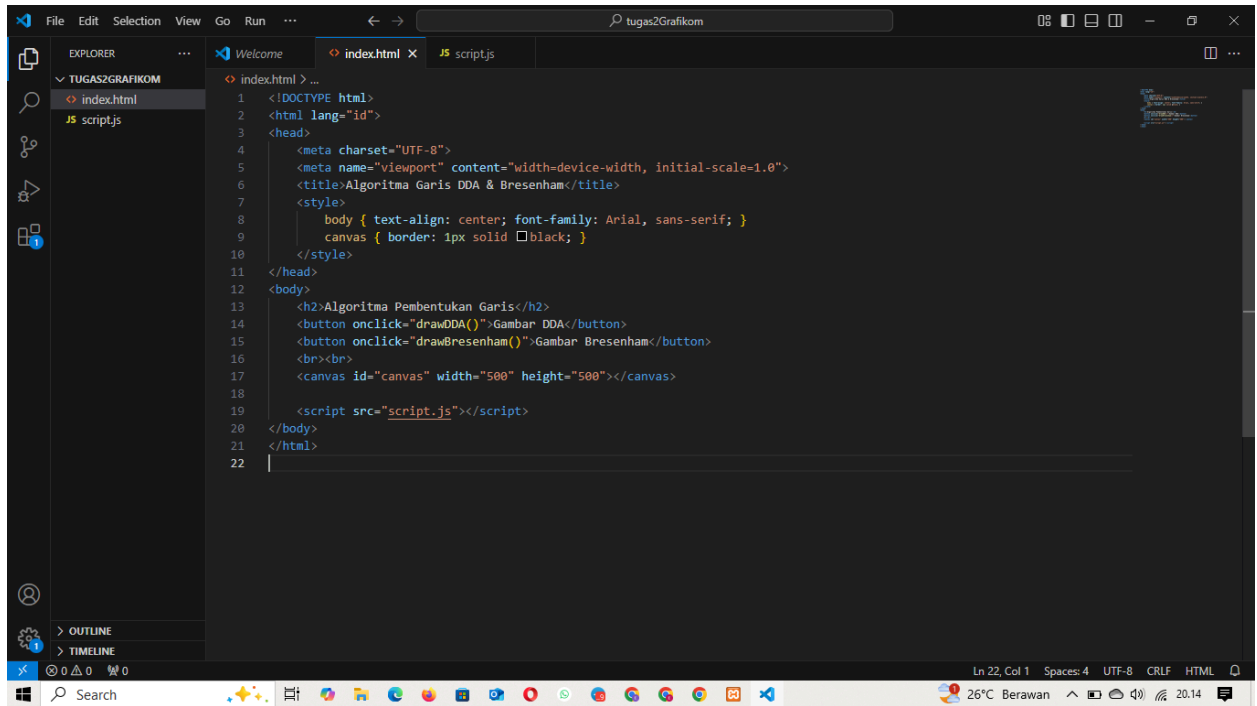


Nama : Eriza Tri Sativa
NPM : 2217051159
Kelas : C

Tugas 2 Grafika Komputer Algoritma Pembentukan Garis

1. Algoritma Bresenham

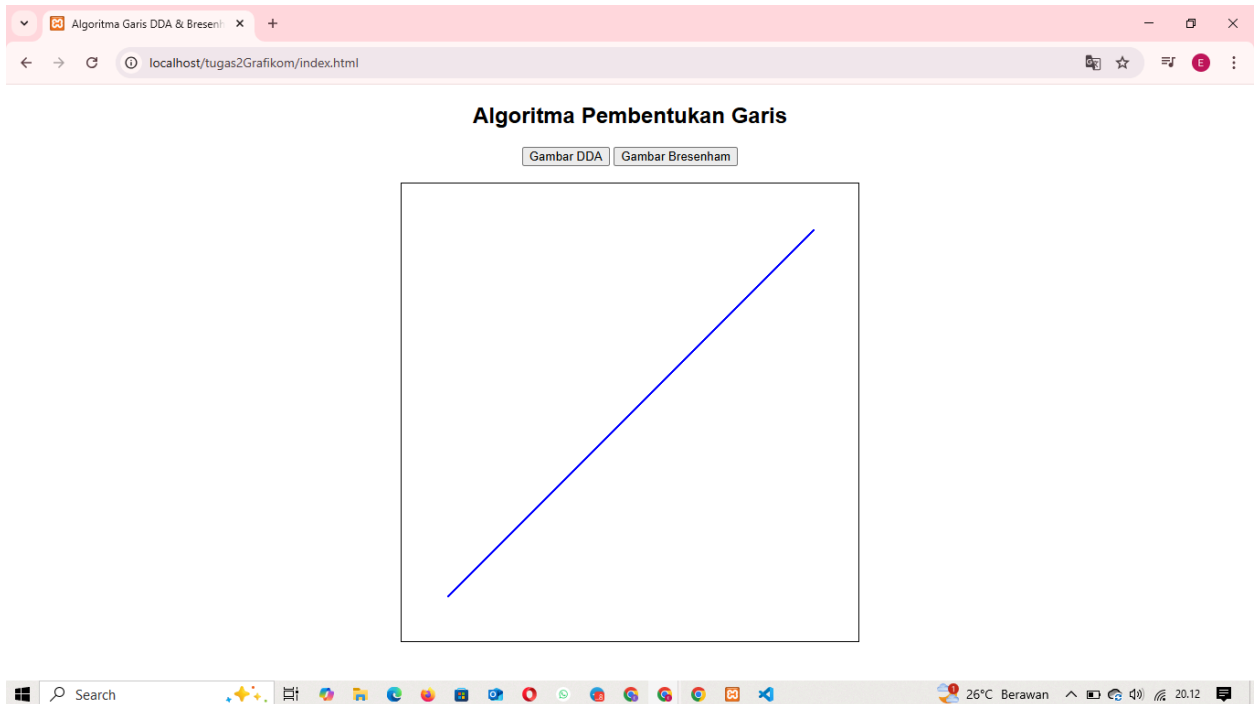
Source code untuk tampilan Pembentukan Garis Algoritma Bresenham



```
1 <!DOCTYPE html>
2 <html lang="id">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>Algoritma Garis DDA & Bresenham</title>
7   <style>
8     body { text-align: center; font-family: Arial, sans-serif; }
9     canvas { border: 1px solid black; }
10  </style>
11 </head>
12 <body>
13   <h2>Algoritma Pembentukan Garis</h2>
14   <button onclick="drawDDA()">Gambar DDA</button>
15   <button onclick="drawBresenham()">Gambar Bresenham</button>
16   <br><br>
17   <canvas id="canvas" width="500" height="500"></canvas>
18
19   <script src="script.js"></script>
20 </body>
21 </html>
22
```

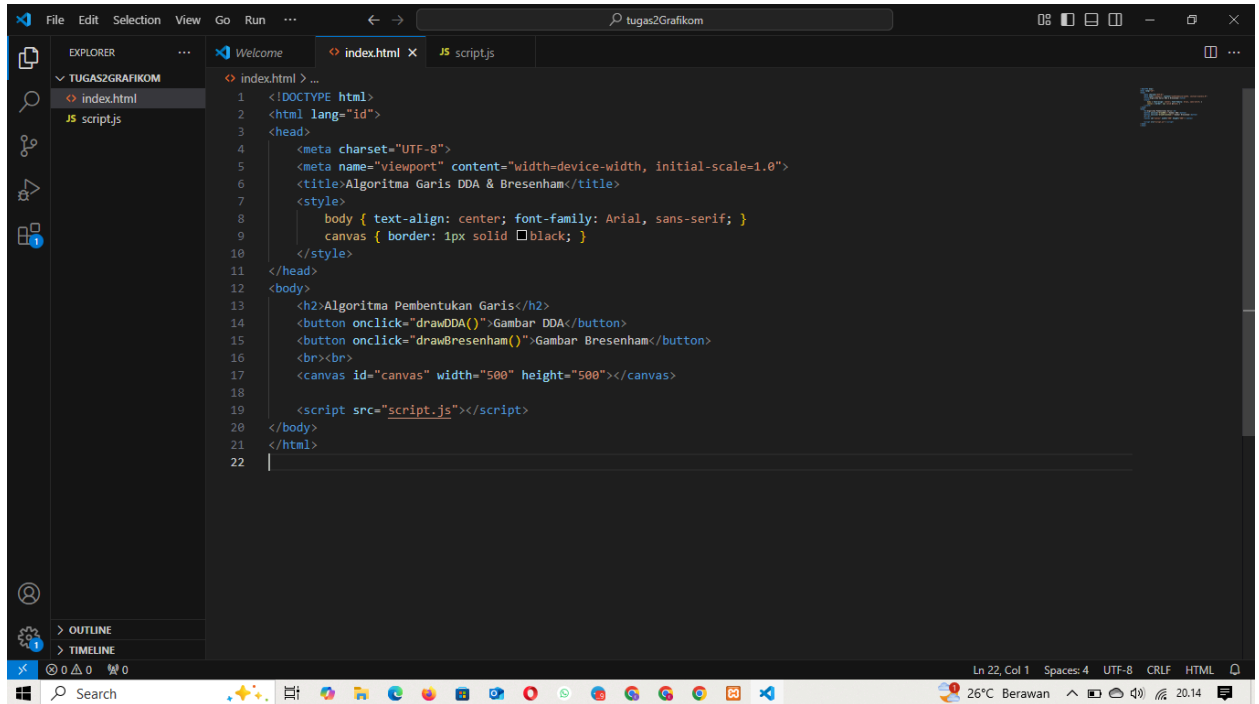
```
File Edit Selection View Go Run ... < -> tugas2Grafikom
EXPLORER
TUGAS2...
index.html
JS scripts
JS scripts > ...
11 function drawDDA() {
21   for (let i = 0; i <= steps; i++) {
22     drawPixel(Math.round(x), Math.round(y), "red");
23     x += Xinc;
24     y += Yinc;
25   }
26 }
27
28 // Algoritma Bresenham
29 function drawBresenham() {
30   ctx.clearRect(0, 0, canvas.width, canvas.height);
31
32   let x0 = 50, y0 = 450, x1 = 450, y1 = 50;
33   let dx = Math.abs(x1 - x0), dy = Math.abs(y1 - y0);
34   let sx = x0 < x1 ? 1 : -1;
35   let sy = y0 < y1 ? 1 : -1;
36   let err = dx - dy;
37
38   while (true) {
39     drawPixel(x0, y0, "blue");
40     if (x0 === x1 && y0 === y1) break;
41
42     let e2 = 2 * err;
43     if (e2 > -dy) { err -= dy; x0 += sx; }
44     if (e2 < dx) { err += dx; y0 += sy; }
45   }
46 }
47
Ln 10, Col 10 Spaces: 4 UTF-8 CRLF JavaScript
Search 26°C Berawan 20.16
```

Hasil tampilan Pembentukan Garis Algoritma Bresenham

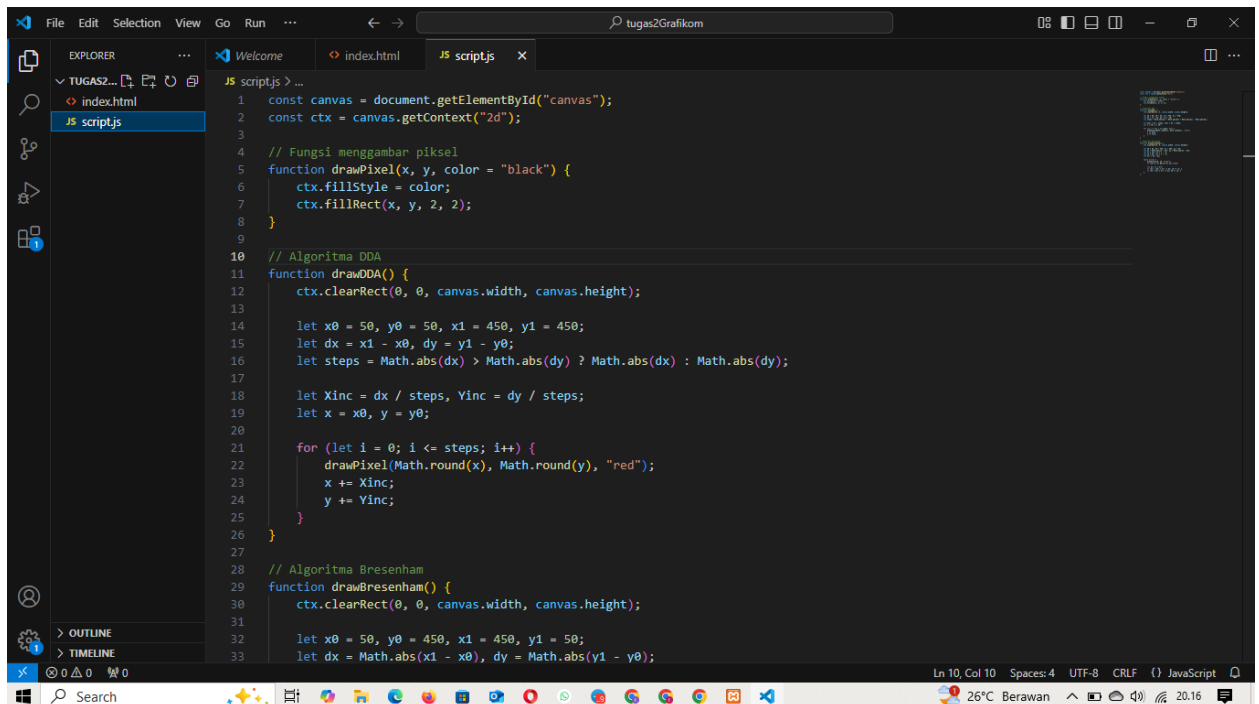


2. Algoritma DDA

Source code untuk tampilan Pembentukan Garis Algoritma DDA



```
1 <!DOCTYPE html>
2 <html lang="id">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>Algoritma Garis DDA & Bresenham</title>
7   <style>
8     body { text-align: center; font-family: Arial, sans-serif; }
9     canvas { border: 1px solid black; }
10  </style>
11 </head>
12 <body>
13   <h2>Algoritma Pembentukan Garis</h2>
14   <button onclick="drawDDA()">Gambar DDA</button>
15   <button onclick="drawBresenham()">Gambar Bresenham</button>
16   <br><br>
17   <canvas id="canvas" width="500" height="500"></canvas>
18
19   <script src="script.js"></script>
20 </body>
21 </html>
22
```



```
1 const canvas = document.getElementById("canvas");
2 const ctx = canvas.getContext("2d");
3
4 // Fungsi menggambar piksel
5 function drawPixel(x, y, color = "black") {
6   ctx.fillStyle = color;
7   ctx.fillRect(x, y, 2, 2);
8 }
9
10 // Algoritma DDA
11 function drawDDA() {
12   ctx.clearRect(0, 0, canvas.width, canvas.height);
13
14   let x0 = 50, y0 = 50, x1 = 450, y1 = 450;
15   let dx = x1 - x0, dy = y1 - y0;
16   let steps = Math.abs(dx) > Math.abs(dy) ? Math.abs(dx) : Math.abs(dy);
17
18   let Xinc = dx / steps, Yinc = dy / steps;
19   let x = x0, y = y0;
20
21   for (let i = 0; i <= steps; i++) {
22     drawPixel(Math.round(x), Math.round(y), "red");
23     x += Xinc;
24     y += Yinc;
25   }
26 }
27
28 // Algoritma Bresenham
29 function drawBresenham() {
30   ctx.clearRect(0, 0, canvas.width, canvas.height);
31
32   let x0 = 50, y0 = 450, x1 = 450, y1 = 50;
33   let dx = Math.abs(x1 - x0), dy = Math.abs(y1 - y0);
34 }
```

Hasil tampilan Pembentukan Garis Algoritma DDA

