**GRAFIKA KOMPUTER**

**“Implementasi Algoritma Pembentukan Kurva Menggunakan HTML”**

****

**Dosen Mata Kuliah :**

Febi Eka Febriansyah,,

M.T. Wartariyus, S.Kom.,M.T.I.

Putut Aji Nalendro, M.Pd.

**Disusun Oleh :**

Nama : Elsa Susanti

NPM : 2413025027

Kelas : 2024A

**PROGRAM STUDI PENDIDIKAN TEKNOLOGI INFORMASI**

**FAKULTAS KEGURUAN DAN ILMU PENDIDIKAN**

**UNIVERSITAS LAMPUNG \**

**2025**

1. **KURVA BEZIER KUBIK**

Kurva Bézier kubik adalah jenis kurva yang sering digunakan dalam grafika komputer untuk membuat lengkungan yang halus dan mudah dikendalikan. Kurva ini terdiri dari empat titik utama, yaitu P₀, P₁, P₂, dan P₃. Titik P₀ dan P₃ adalah titik awal dan akhir dari kurva, sedangkan P₁ dan P₂ berfungsi untuk mengatur arah dan bentuk lengkungan kurva. Namun, titik kontrol ini tidak dilalui oleh kurva, mereka hanya membantu membentuk jalan yang diambil oleh kurva tersebut.

Secara sederhana, kurva Bézier kubik dapat dipahami dengan memvisualisasikan bagaimana perubahan posisi titik P₁ dan P₂ akan mengubah bentuk lengkungan kurva yang menghubungkan titik P₀ dan P₃. Ini memungkinkan kita untuk membuat kurva yang sangat fleksibel dan mudah dikendalikan hanya dengan memindahkan beberapa titik.

Matematika di balik kurva ini cukup sederhana, meskipun hasilnya bisa sangat halus dan kompleks. Persamaan yang digunakan untuk menggambarkan kurva Bézier kubik adalah:

B(t) = (1−t)³P₀ + 3(1−t)²tP₁ + 3(1−t)t²P₂ + t³P₃,

Di sini, t adalah parameter yang berjalan dari 0 hingga 1. Ketika t bernilai 0, kurva berada di titik P₀ (awal), dan saat t bernilai 1, kurva berada di titik P₃ (akhir). Sementara itu, titik P₁ dan P₂ berfungsi untuk mengubah bentuk kurva di sepanjang jalan tersebut.

Kurva Bézier kubik sangat populer dalam desain grafis, terutama untuk membuat desain vektor yang halus dan kompleks, seperti logo dan ilustrasi digital.

Di dunia pemrograman web, kurva ini juga digunakan dalam animasi CSS, misalnya, untuk membuat transisi yang halus antara satu keadaan dan keadaan lainnya. Jadi, kurva Bézier kubik tidak hanya berguna dalam grafika komputer, tetapi juga dalam membuat antarmuka pengguna yang lebih dinamis dan responsif.

1. **IMPLEMENTASI PADA HTML**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Elsa Susanti  - Kurva Bezier Cubic – Pohon, Awan, dan Semak</title>

  <style>

    canvas {

      border: 2px solid #87CEEB;

    }

    h1 {

      text-align: center;

    }

  </style>

</head>

<body>

  <h1>POHON, AWAN, DAN SEMAK</h1>

  <canvas id="canvas" width="800" height="800"></canvas>

  <a href="index.html">Kembali ke home</a>

  <script>

    const canvas = document.getElementById("canvas");

    const ctx = canvas.getContext("2d");

    // Fungsi menggambar kurva Bezier

    function drawBezierCurve(ctx, points, color) {

      ctx.beginPath();

      ctx.moveTo(points[0].x, points[0].y);

      ctx.strokeStyle = color;

      ctx.fillStyle = color;

      for (let i = 1; i < points.length - 2; i += 3) {

        ctx.bezierCurveTo(

          points[i].x, points[i].y,

          points[i + 1].x, points[i + 1].y,

          points[i + 2].x, points[i + 2].y

        );

      }

      ctx.fill();

      ctx.closePath();

      ctx.stroke();

    }

    // Daun 1

    const Daun1\_points = [

      {x: 280, y: 320}, {x: 190, y: 340}, {x: 200, y: 225},

      {x: 240, y: 200}, {x: 280, y: 190}, {x: 280, y: 220},

      {x: 280, y: 205}, {x: 270, y: 200}, {x: 360, y: 120},

      {x: 380, y: 220}, {x: 510, y: 210}, {x: 410, y: 410},

      {x: 295, y: 310}, {x: 250, y: 145}, {x: 370, y: 110},

      {x: 400, y: 180}, {x: 430, y: 180}, {x: 460, y: 250},

      {x: 380, y: 300}, {x: 340, y: 310}, {x: 290, y: 300},

      {x: 250, y: 320}, {x: 280, y: 190}, {x: 200, y: 150},

      {x: 370, y: 160}, {x: 240, y: 90}

    ];

    drawBezierCurve(ctx, Daun1\_points, "green");

    // Daun 2

    const Daun2\_points = Daun1\_points.map(point => ({

      x: point.x - 200,

      y: point.y - 120

    }));

    drawBezierCurve(ctx, Daun2\_points, "green");

    // Semak 1

    const Semak1\_points = [

      {x: 150, y: 410}, {x: 190, y: 320}, {x: 200, y: 430},

      {x: 210, y: 370}, {x: 240, y: 390}, {x: 255, y: 400},

      {x: 250, y: 375}, {x: 370, y: 440}, {x: 150, y: 400},

      {x: 150, y: 410}

    ];

    drawBezierCurve(ctx, Semak1\_points, "green");

    // Semak 2

    const Semak2\_points = Semak1\_points.map(point => ({

      x: point.x + 190,

      y: point.y - 10

    }));

    drawBezierCurve(ctx, Semak2\_points, "green");

    // Semak 3

    const Semak3\_points = Semak1\_points.map(point => ({

      x: point.x - 150,

      y: point.y - 80

    }));

    drawBezierCurve(ctx, Semak3\_points, "green");

    // Pohon 1

    const branchPoints1 = [

      {x: 310, y: 480}, {x: 310, y: 320}, {x: 310, y: 310},

      {x: 300, y: 290}, {x: 180, y: 185}, {x: 330, y: 345},

      {x: 325, y: 245}, {x: 330, y: 360}, {x: 460, y: 190},

      {x: 350, y: 310}, {x: 350, y: 470}, {x: 350, y: 370},

      {x: 350, y: 480}

    ];

    drawBezierCurve(ctx, branchPoints1, "tan");

    // Pohon 2

    const branchPoints2 = branchPoints1.map(point => ({

      x: point.x - 200,

      y: point.y - 120

    }));

    drawBezierCurve(ctx, branchPoints2, "tan");

    // Awan 1

    const cloud1\_points = [

      {x: 370, y: 100}, {x: 265, y: 100}, {x: 200, y: 105},

      {x: 268, y: 75}, {x: 250, y: 80}, {x: 300, y: 30},

      {x: 330, y: 70}, {x: 370, y: 70}, {x: 370, y: 100}

    ];

    drawBezierCurve(ctx, cloud1\_points, "skyblue");

    // Awan 2

    const cloud2\_points = [

      {x: 465, y: 150}, {x: 350, y: 145}, {x: 300, y: 155},

      {x: 368, y: 125}, {x: 350, y: 135}, {x: 385, y: 80},

      {x: 430, y: 120}, {x: 400, y: 120}, {x: 450, y: 150}

    ];

    drawBezierCurve(ctx, cloud2\_points, "skyblue");

    // Resize canvas (optional)

    window.addEventListener("resize", () => {

      canvas.width = window.innerWidth;

      canvas.height = window.innerHeight;

      // (Redraw bisa ditambahkan di sini kalau mau responsif)

    });

  </script>

</body>

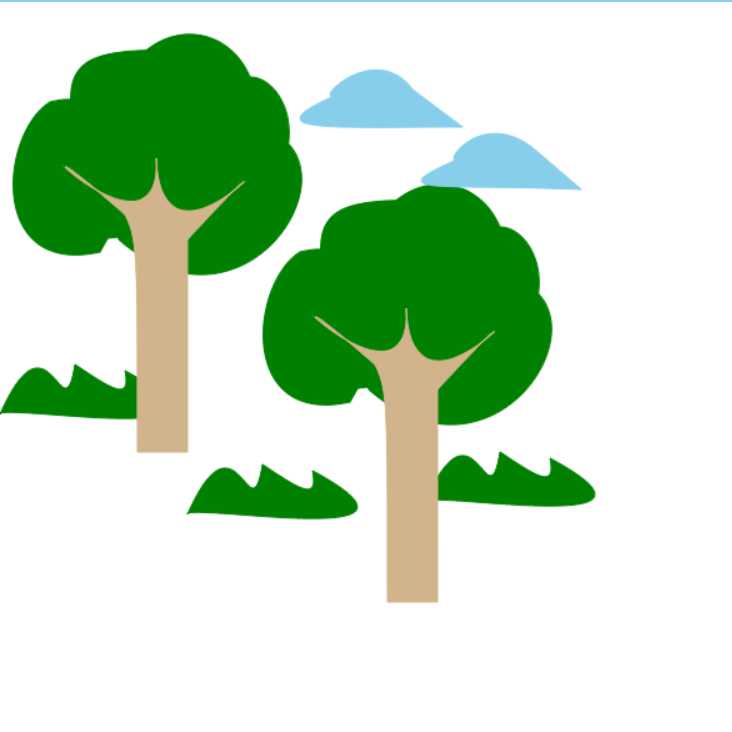
</html

Dalam program ini, kita akan melihat tampilan grafis yang berisi objek-objek seperti pohon, awan, dan semak yang digambar menggunakan algoritma kurva Bézier kubik. Berikut adalah hasil yang dapat diharapkan dari program ini:

1. Pohon akan terlihat dengan batang yang digambar menggunakan garis dan daun yang digambar menggunakan kurva Bézier untuk menciptakan bentuk lengkungan yang alami dan halus.
2. Awan digambar menggunakan beberapa kurva Bézier yang membentuk tampilan awan dengan warna biru langit, memberikan efek yang lembut dan mengalir.
3. Semak dibuat dengan menggunakan kurva Bézier untuk menghasilkan bentuk semak yang lebat dan alami.
4. Interaktivitas Responsif: Jika ukuran jendela browser diubah, ukuran canvas akan menyesuaikan secara otomatis untuk memastikan gambar tetap proporsional dan tampilan tetap terjaga meskipun ukuran layar berubah.

Secara keseluruhan, hasil yang diperoleh adalah sebuah gambar alam digital yang terdiri dari pohon, semak, dan awan yang semuanya dibentuk secara halus menggunakan kurva Bézier kubik.

**OUTPUT/HASIL**



1. **KESIMPULAN**

Dalam implementasi algoritma kurva Bézier kubik menggunakan HTML dan canvas, kita dapat menghasilkan objek grafis yang halus dan fleksibel, seperti pohon, awan, dan semak. Kurva Bézier kubik memungkinkan pembuatan lengkungan yang mudah dikendalikan hanya dengan mengubah posisi beberapa titik. Ini memudahkan dalam pembuatan desain grafis dan animasi yang responsif di web.

Dengan menggunakan elemen <canvas> dan JavaScript, kita dapat menggambar objek-objek tersebut secara dinamis dan membuat tampilan yang menyesuaikan dengan ukuran layar. Selain itu, kurva Bézier kubik juga banyak digunakan dalam desain grafis dan animasi web untuk transisi yang halus dan efek visual yang lebih menarik.

Secara keseluruhan, penerapan kurva Bézier kubik membantu menciptakan desain yang lebih fleksibel dan interaktif, membuatnya menjadi alat yang sangat berguna dalam grafika komputer dan pengembangan web.