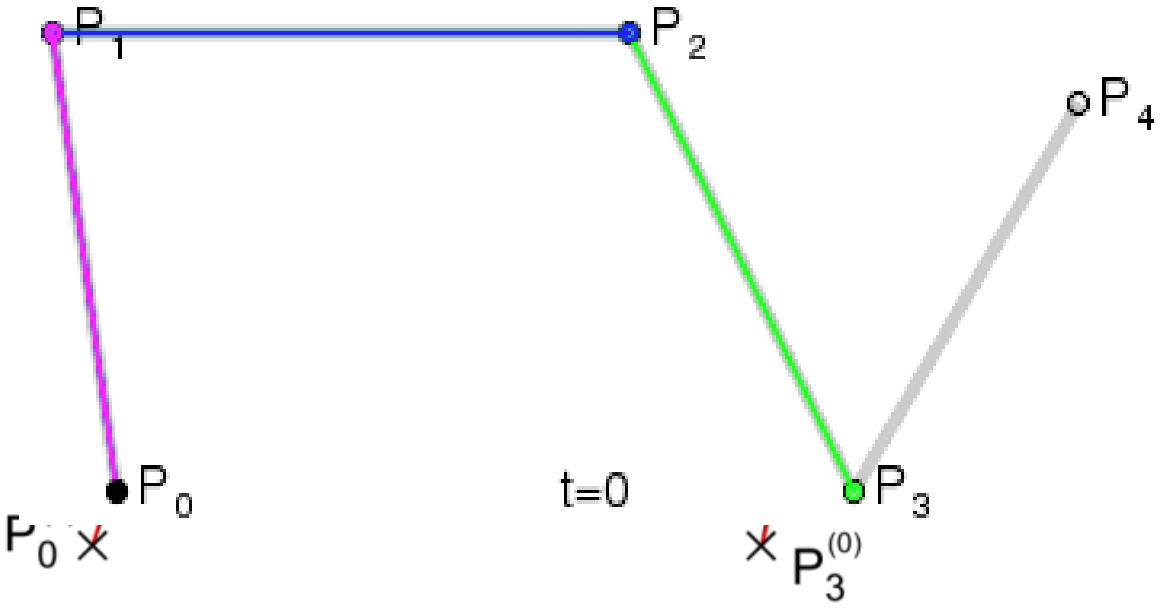
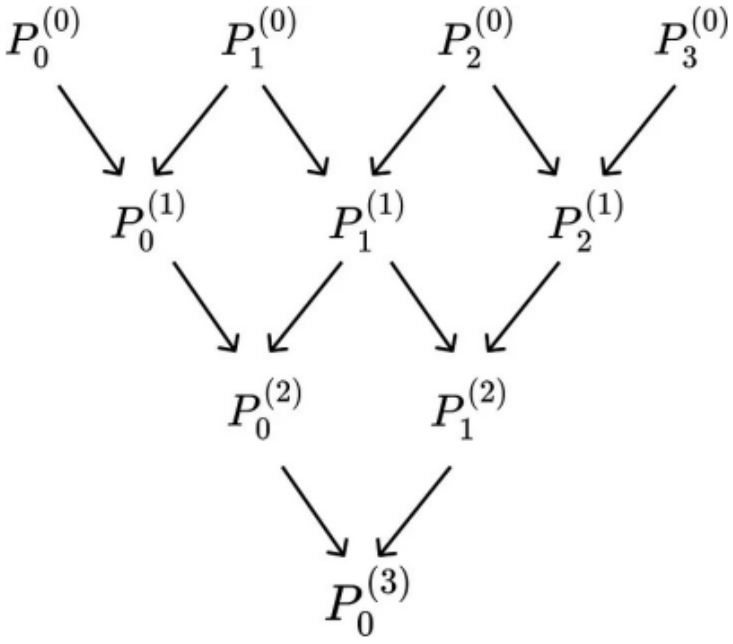
Simulating De'Casteljau's Algorithm in Canvas

De Casteljau's algorithm



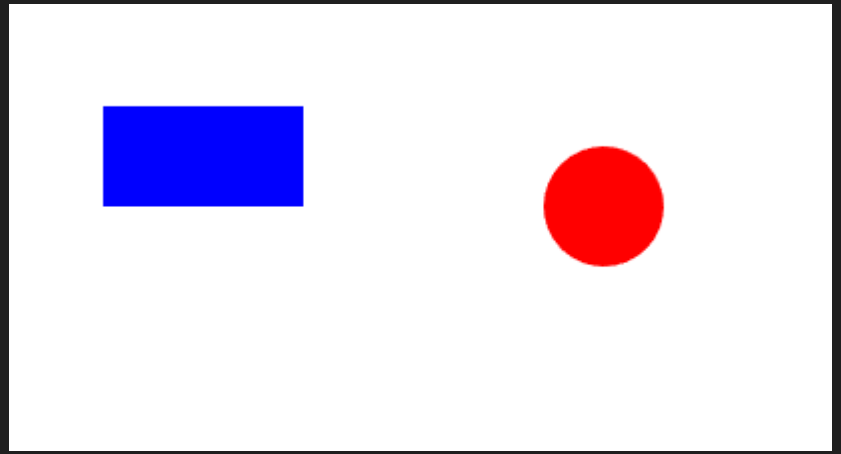


$$P_i^{(j)} = (1-t)P_i^{j-1} + tP_{i+1}^{j-1}$$

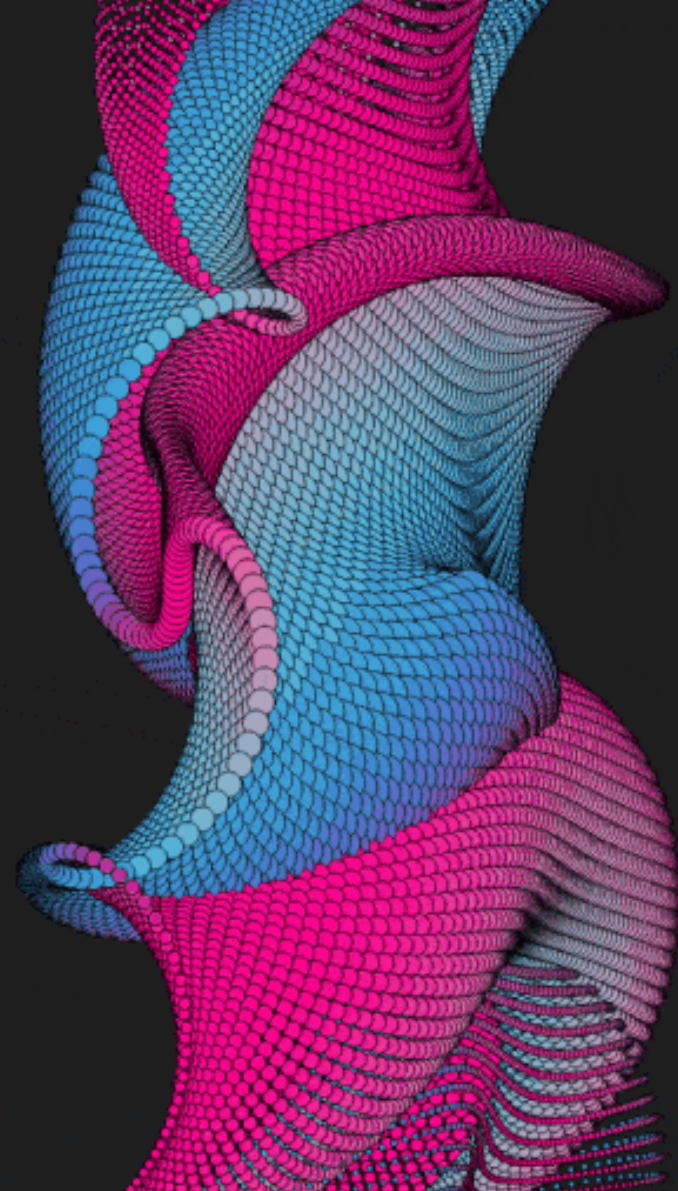
HTML5

< canvas >

```html
<canvas id="myCanvas" width="400" height="200"></canvas>
<script>
  // Get the canvas element
  const canvas = document.getElementById("myCanvas");
  // Get the 2D drawing context
  const ctx = canvas.getContext("2d");
  // Example 1: Drawing a Rectangle
  ctx.fillStyle = "blue"; // Set fill color to blue
  ctx.fillRect(50, 50, 100, 50); // Draw a filled rectangle
  // Example 2: Drawing a Circle
  ctx.fillStyle = "red"; // Set fill color to red
  ctx.beginPath();
  ctx.arc(300, 100, 30, 0, 2 * Math.PI); // Draw a circle
  ctx.fill(); // Fill the circle
</script>
```

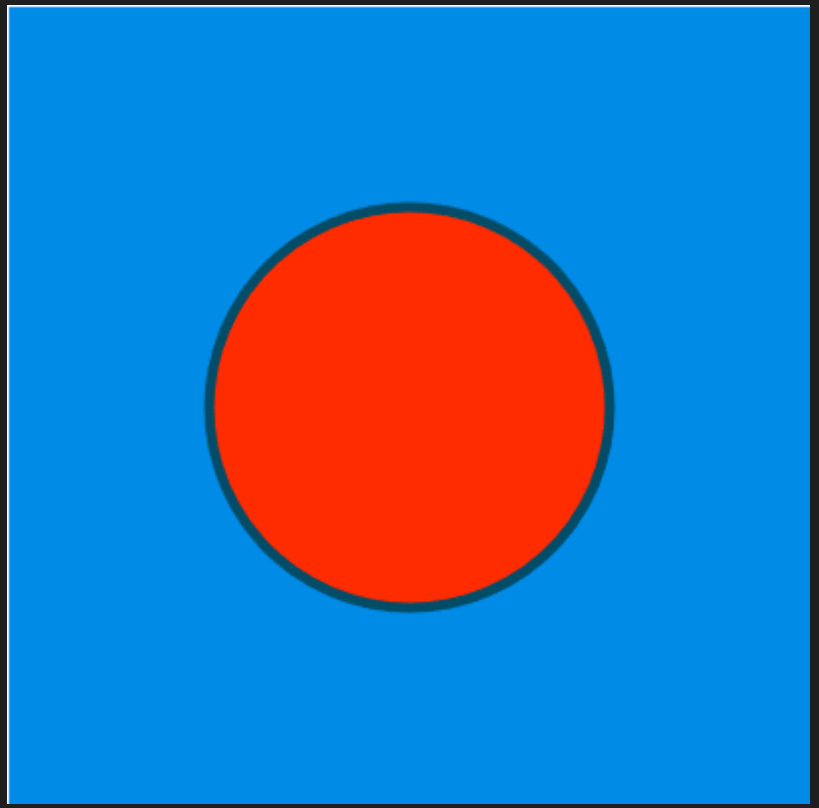


p5*



```
<script src="https://cdn.jsdelivr.net/npm/p5@1.9.1/lib/p5.js"></script>
<script>
  function setup() {
    createCanvas(400, 400);
    colorMode(RGB);
  }

  function draw() {
    background(0, 140, 230);
    fill(255, 44, 0);
    stroke(2, 76, 104);
    strokeWeight(5);
    ellipse(200, 200, 200, 200);
  }
</script>
```



P5.js vs Canvas API

Pros of p5js

- Beginner-friendly
- Large community
- Resources
- Documentation
- Convenience
- Hides complexity

Cons of p5js

- Hides complexity!
- Size
- Speed
- Some bad practices
- Compatibility
- Prototyping

Pros of HTML Canvas

- Speed
- Deeper understanding
- Power
- Future-proof
- Documentiation

Cons of HTML Canvas

- Difficulty
- Verbosity
- Less focused resources

<Let’s Code />