

# DESENHO TÉCNICO E COMPUTAÇÃO GRÁFICA

Ms Márcio J Morais

**GRANTiETE**  
F A C U L D A D E

Engenharia da Computação

# **FACULDADE GRAN TIETÊ**

Av. 15 de Novembro, 125 - Centro - Barra Bonita/SP - CEP: 17340-047

(14) 3642-3219

[contato@grantiete.com.br](mailto:contato@grantiete.com.br)

<https://grantiete.com.br/>

## **Professor Mestre Márcio Jesus de Moraes**

[marciojmoraes@gmail.com](mailto:marciojmoraes@gmail.com)

## **Bacharelado em Engenharia da Computação**

### **Desenho Técnico e Expressão Gráfica**

**1.º Bimestre – 04/08 a 10/10**

**2.º Bimestre – 13/10 a 19/12**

### **Turmas/Termos**

Eng. Comp.: 1.º, 2.º, 3.º; 4.º, 5.º, 6.º, 8.º, 9.º e 10.º

Eng. Civil: 4.º, 5.º e 6.º

**Terças-feiras - 19h00 – 22h30 - Sala Allan Turing**

# AULA 03 - Construções Geométricas

## 1. PONTO - Primeiro Passo

### Conceito

Um ponto é a menor unidade geométrica possível.

- Não tem dimensão (largura = 0, altura = 0)
- Apenas indica POSIÇÃO
- É representado por coordenadas (x, y)

Vamos desenhar um ponto no canvas.

```
<!DOCTYPE html>
<html>
<head>
  <title>1. Um Ponto</title>
</head>
<body>
  <h2>Desenhando UM ponto</h2>
  <canvas id="canvas" width="400" height="300" style="border: 1px solid black;"></canvas>

  <script>
    // Pega o canvas e o contexto de desenho
    const canvas = document.getElementById('canvas');
    const ctx = canvas.getContext('2d');

    // Define a cor de preenchimento como vermelha
    ctx.fillStyle = 'red';

    // Desenha um círculo (ponto) na posição x=100, y=100 com raio 5
    ctx.beginPath();
    ctx.arc(100, 100, 5, 0, 2 * Math.PI);
    ctx.fill();
  </script>
</body>
</html>
```

**Exercício 01:** Modifique o código para colocar o ponto na posição (200, 150).

## 2. DOIS PONTOS

Agora vamos desenhar dois pontos em posições diferentes.

```
<!DOCTYPE html>
<html>
<head>
  <title>2. Dois Pontos</title>
</head>
<body>
  <h2>Desenhando DOIS pontos</h2>
  <canvas id="canvas" width="400" height="300" style="border: 1px solid black;"></canvas>

  <script>
    const canvas = document.getElementById('canvas');
    const ctx = canvas.getContext('2d');

    // Primeiro ponto - vermelho
    ctx.fillStyle = 'red';
    ctx.beginPath();
    ctx.arc(100, 100, 5, 0, 2 * Math.PI);
    ctx.fill();

    // Segundo ponto - azul
    ctx.fillStyle = 'blue';
    ctx.beginPath();
    ctx.arc(250, 150, 5, 0, 2 * Math.PI);
    ctx.fill();
  </script>
</body>
</html>
```

**Exercício 02:** Adicione um terceiro ponto verde na posição (300, 200).

### 3. TRÊS PONTOS COM NOMES

Vamos desenhar três pontos e dar nomes a eles (A, B, C).

```
<!DOCTYPE html>
<html>
<head>
  <title>3. Três Pontos com Nomes</title>
</head>
<body>
  <h2>Três pontos: A, B e C</h2>
  <canvas id="canvas" width="400" height="300" style="border: 1px solid black;"></canvas>

  <script>
    const canvas = document.getElementById('canvas');
    const ctx = canvas.getContext('2d');

    // Ponto A
    ctx.fillStyle = 'red';
    ctx.beginPath();
    ctx.arc(100, 100, 5, 0, 2 * Math.PI);
    ctx.fill();
    ctx.fillStyle = 'black';
    ctx.font = '16px Arial';
    ctx.fillText('A', 110, 105);

    // Ponto B
    ctx.fillStyle = 'blue';
    ctx.beginPath();
    ctx.arc(250, 150, 5, 0, 2 * Math.PI);
    ctx.fill();
    ctx.fillStyle = 'black';
    ctx.fillText('B', 260, 155);

    // Ponto C
    ctx.fillStyle = 'green';
    ctx.beginPath();
    ctx.arc(180, 200, 5, 0, 2 * Math.PI);
    ctx.fill();
    ctx.fillStyle = 'black';
    ctx.fillText('C', 190, 205);
  </script>
</body>
</html>
```

**Exercício 03:** Mude as posições dos pontos para formar um triângulo mais simétrico.

## 4. PRIMEIRO SEGMENTO

Um segmento é uma linha reta entre dois pontos. Vamos conectar dois pontos.

```
<!DOCTYPE html>
<html>
<head>
  <title>4. Primeiro Segmento</title>
</head>
<body>
  <h2>Meu primeiro segmento AB</h2>
  <canvas id="canvas" width="400" height="300" style="border: 1px solid black;"></canvas>

  <script>
    const canvas = document.getElementById('canvas');
    const ctx = canvas.getContext('2d');

    // Coordenadas dos pontos
    const pontoA = {x: 100, y: 100};
    const pontoB = {x: 300, y: 150};

    // Desenha a linha (segmento)
    ctx.strokeStyle = 'black';
    ctx.lineWidth = 2;
    ctx.beginPath();
    ctx.moveTo(pontoA.x, pontoA.y); // Move para ponto A
    ctx.lineTo(pontoB.x, pontoB.y); // Desenha linha até ponto B
    ctx.stroke();

    // Desenha os pontos
    ctx.fillStyle = 'red';
    ctx.beginPath();
    ctx.arc(pontoA.x, pontoA.y, 5, 0, 2 * Math.PI);
    ctx.fill();

    ctx.beginPath();
    ctx.arc(pontoB.x, pontoB.y, 5, 0, 2 * Math.PI);
    ctx.fill();

    // Nomes dos pontos
    ctx.fillStyle = 'black';
    ctx.font = '16px Arial';
    ctx.fillText('A', pontoA.x - 15, pontoA.y - 10);
    ctx.fillText('B', pontoB.x + 10, pontoB.y - 10);
  </script>
</body>
</html>
```

**Exercício 04:** Crie um segmento vertical conectando os pontos (200, 50) e (200, 250).



## 5. DOIS SEGMENTOS

Agora vamos desenhar dois segmentos diferentes no mesmo canvas.

```
<!DOCTYPE html>
<html>
<head>
  <title>5. Dois Segmentos</title>
</head>
<body>
  <h2>Dois segmentos: AB e CD</h2>
  <canvas id="canvas" width="400" height="300" style="border: 1px solid black;"></canvas>

  <script>
    const canvas = document.getElementById('canvas');
    const ctx = canvas.getContext('2d');

    // Pontos do primeiro segmento
    const A = {x: 50, y: 100};
    const B = {x: 200, y: 120};

    // Pontos do segundo segmento
    const C = {x: 100, y: 200};
    const D = {x: 350, y: 180};

    // Primeiro segmento AB (azul)
    ctx.strokeStyle = 'blue';
    ctx.lineWidth = 3;
    ctx.beginPath();
    ctx.moveTo(A.x, A.y);
    ctx.lineTo(B.x, B.y);
    ctx.stroke();

    // Segundo segmento CD (verde)
    ctx.strokeStyle = 'green';
    ctx.lineWidth = 3;
    ctx.beginPath();
    ctx.moveTo(C.x, C.y);
    ctx.lineTo(D.x, D.y);
    ctx.stroke();

    // Desenha todos os pontos
    ctx.fillStyle = 'red';
    [A, B, C, D].forEach(ponto => {
      ctx.beginPath();
      ctx.arc(ponto.x, ponto.y, 4, 0, 2 * Math.PI);
      ctx.fill();
    });

    // Nomes dos pontos
    ctx.fillStyle = 'black';
    ctx.font = '14px Arial';
    ctx.fillText('A', A.x - 15, A.y - 10);
    ctx.fillText('B', B.x + 10, B.y - 10);
    ctx.fillText('C', C.x - 15, C.y + 20);
    ctx.fillText('D', D.x + 10, D.y + 20);
  </script>
</body>
</html>
```

**Exercício 05:** Adicione um terceiro segmento EF com pontos de sua escolha.

## 6. RETAS PARALELAS SIMPLES

Retas paralelas nunca se encontram. Vamos desenhar duas retas paralelas horizontais.

```
<!DOCTYPE html>
<html>
<head>
  <title>6. Retas Paralelas</title>
</head>
<body>
  <h2>Duas retas paralelas</h2>
  <canvas id="canvas" width="400" height="300" style="border: 1px solid black;"></canvas>

  <script>
    const canvas = document.getElementById('canvas');
    const ctx = canvas.getContext('2d');

    // Primeira reta (horizontal superior)
    ctx.strokeStyle = 'blue';
    ctx.lineWidth = 2;
    ctx.beginPath();
    ctx.moveTo(50, 100); // Início da linha
    ctx.lineTo(350, 100); // Fim da linha
    ctx.stroke();

    // Segunda reta (horizontal inferior, paralela à primeira)
    ctx.strokeStyle = 'red';
    ctx.lineWidth = 2;
    ctx.beginPath();
    ctx.moveTo(50, 200); // Mesma direção, 100px abaixo
    ctx.lineTo(350, 200);
    ctx.stroke();

    // Texto explicativo
    ctx.fillStyle = 'black';
    ctx.font = '12px Arial';
    ctx.fillText('Reta 1 (azul)', 60, 90);
    ctx.fillText('Reta 2 (vermelha)', 60, 190);
    ctx.fillText('Distância: 100px', 60, 150);

    // Linha tracejada mostrando a distância
    ctx.strokeStyle = 'gray';
    ctx.setLineDash([5, 5]);
    ctx.beginPath();
    ctx.moveTo(100, 100);
    ctx.lineTo(100, 200);
    ctx.stroke();
  </script>
</body>
</html>
```

**Exercício 06:** Crie duas retas paralelas verticais com 150px de distância entre elas.



## 7. RETAS PERPENDICULARES SIMPLES

Retas perpendiculares se cruzam formando ângulo de 90°.

```
<!DOCTYPE html>
<html>
<head>
  <title>7. Retas Perpendiculares</title>
</head>
<body>
  <h2>Retas perpendiculares (90°)</h2>
  <canvas id="canvas" width="400" height="300" style="border: 1px solid black;"></canvas>

  <script>
    const canvas = document.getElementById('canvas');
    const ctx = canvas.getContext('2d');

    // Ponto de intersecção
    const centro = {x: 200, y: 150};

    // Reta horizontal
    ctx.strokeStyle = 'blue';
    ctx.lineWidth = 3;
    ctx.beginPath();
    ctx.moveTo(50, centro.y);
    ctx.lineTo(350, centro.y);
    ctx.stroke();

    // Reta vertical (perpendicular à horizontal)
    ctx.strokeStyle = 'red';
    ctx.lineWidth = 3;
    ctx.beginPath();
    ctx.moveTo(centro.x, 50);
    ctx.lineTo(centro.x, 250);
    ctx.stroke();

    // Marca o ponto de intersecção
    ctx.fillStyle = 'green';
    ctx.beginPath();
    ctx.arc(centro.x, centro.y, 5, 0, 2 * Math.PI);
    ctx.fill();

    // Desenha o símbolo de ângulo reto (quadrado)
    ctx.strokeStyle = 'green';
    ctx.lineWidth = 2;
    ctx.strokeRect(centro.x - 10, centro.y - 10, 20, 20);

    // Texto
    ctx.fillStyle = 'black';
    ctx.font = '12px Arial';
    ctx.fillText('90°', centro.x + 15, centro.y - 15);
  </script>
</body>
</html>
```

**Exercício 07:** Mova o ponto de intersecção para (150, 100) e ajuste as retas.

## 8. DIVIDINDO UM SEGMENTO AO MEIO

Vamos encontrar o ponto médio de um segmento.

```
<!DOCTYPE html>
<html>
<head>
  <title>8. Ponto Médio</title>
</head>
<body>
  <h2>Dividindo segmento ao meio</h2>
  <canvas id="canvas" width="400" height="300" style="border: 1px solid black;"></canvas>

  <script>
    const canvas = document.getElementById('canvas');
    const ctx = canvas.getContext('2d');

    // Pontos extremos do segmento
    const A = {x: 100, y: 100};
    const B = {x: 300, y: 200};

    // Calcula o ponto médio
    const meio = {
      x: (A.x + B.x) / 2, // Média das coordenadas x
      y: (A.y + B.y) / 2 // Média das coordenadas y
    };

    // Desenha o segmento
    ctx.strokeStyle = 'black';
    ctx.lineWidth = 3;
    ctx.beginPath();
    ctx.moveTo(A.x, A.y);
    ctx.lineTo(B.x, B.y);
    ctx.stroke();

    // Desenha os pontos extremos
    ctx.fillStyle = 'blue';
    ctx.beginPath();
    ctx.arc(A.x, A.y, 5, 0, 2 * Math.PI);
    ctx.fill();
    ctx.beginPath();
    ctx.arc(B.x, B.y, 5, 0, 2 * Math.PI);
    ctx.fill();

    // Desenha o ponto médio
    ctx.fillStyle = 'red';
    ctx.beginPath();
    ctx.arc(meio.x, meio.y, 6, 0, 2 * Math.PI);
    ctx.fill();

    // Nomes dos pontos
    ctx.fillStyle = 'black';
    ctx.font = '14px Arial';
    ctx.fillText('A', A.x - 15, A.y - 10);
    ctx.fillText('B', B.x + 10, B.y + 20);
    ctx.fillText('M (meio)', meio.x + 10, meio.y - 10);
  </script>
</body>
</html>
```

**Exercício 08:** Calcule e marque o ponto médio entre (50, 50) e (350, 250).

## 9. TRIÂNGULO SIMPLES

Vamos construir um triângulo conectando três pontos.

```
<!DOCTYPE html>
<html>
<head>
  <title>9. Triângulo</title>
</head>
<body>
  <h2>Meu primeiro triângulo</h2>
  <canvas id="canvas" width="400" height="300" style="border: 1px solid black;"></canvas>

  <script>
    const canvas = document.getElementById('canvas');
    const ctx = canvas.getContext('2d');

    // Três vértices do triângulo
    const A = {x: 200, y: 50}; // Vértice superior
    const B = {x: 100, y: 200}; // Vértice inferior esquerdo
    const C = {x: 300, y: 200}; // Vértice inferior direito

    // Desenha o triângulo
    ctx.strokeStyle = 'blue';
    ctx.lineWidth = 2;
    ctx.beginPath();
    ctx.moveTo(A.x, A.y); // Vai para A
    ctx.lineTo(B.x, B.y); // Linha A -> B
    ctx.lineTo(C.x, C.y); // Linha B -> C
    ctx.lineTo(A.x, A.y); // Linha C -> A (fecha o triângulo)
    ctx.stroke();

    // Desenha os vértices
    ctx.fillStyle = 'red';
    [A, B, C].forEach(ponto => {
      ctx.beginPath();
      ctx.arc(ponto.x, ponto.y, 5, 0, 2 * Math.PI);
      ctx.fill();
    });

    // Nomes dos vértices
    ctx.fillStyle = 'black';
    ctx.font = '16px Arial';
    ctx.fillText('A', A.x - 10, A.y - 10);
    ctx.fillText('B', B.x - 20, B.y + 20);
    ctx.fillText('C', C.x + 10, C.y + 20);
  </script>
</body>
</html>
```

**Exercício 09:** Crie um triângulo com vértices em (150, 80), (80, 220) e (220, 220).

## 10. CÍRCULO SIMPLES

Vamos desenhar um círculo com centro e raio definidos.

```
<!DOCTYPE html>
<html>
<head>
  <title>10. Círculo</title>
</head>
<body>
  <h2>Desenhando um círculo</h2>
  <canvas id="canvas" width="400" height="300" style="border: 1px solid black;"></canvas>

  <script>
    const canvas = document.getElementById('canvas');
    const ctx = canvas.getContext('2d');

    // Centro e raio do círculo
    const centro = {x: 200, y: 150};
    const raio = 80;

    // Desenha o círculo (apenas contorno)
    ctx.strokeStyle = 'blue';
    ctx.lineWidth = 3;
    ctx.beginPath();
    ctx.arc(centro.x, centro.y, raio, 0, 2 * Math.PI);
    ctx.stroke();

    // Marca o centro
    ctx.fillStyle = 'red';
    ctx.beginPath();
    ctx.arc(centro.x, centro.y, 4, 0, 2 * Math.PI);
    ctx.fill();

    // Desenha o raio
    ctx.strokeStyle = 'red';
    ctx.lineWidth = 2;
    ctx.beginPath();
    ctx.moveTo(centro.x, centro.y);
    ctx.lineTo(centro.x + raio, centro.y);
    ctx.stroke();

    // Texto
    ctx.fillStyle = 'black';
    ctx.font = '14px Arial';
    ctx.fillText('Centro', centro.x + 10, centro.y - 10);
    ctx.fillText(`Raio = ${raio}px`, centro.x + 20, centro.y + 10);
  </script>
</body>
</html>
```

**Exercício 10:** Crie um círculo com centro em (150, 100) e raio de 60 pixels.