

# **COMPUTAÇÃO GRÁFICA PARA INTERNET**

**Formas vetoriais com CANVAS  
Construção com base no plano  
cartesiano  $(x,y)$  de imagens 2D.**

# CANVAS

É uma tag que possibilita criar, em uma página HTML, uma área para renderização de formas geométricas

# EXEMPLO 01

000	100	200	300	400	500	600
100						
200						
300						
400						
500						
600						

```
ctx.moveTo(100,100); // origem  
ctx.lineTo(200,200); // destino  
ctx.moveTo(200,100); // origem  
ctx.lineTo(100,200); // destino
```

No Caso do X, os vetores não são contínuos, portanto utiliza-se dois pontos de origem(**moveTo**) e dois pontos de destino(**lineTo**).

## ÁREA DE RENDERIZAÇÃO

```
<canvas id='testeCanvas' width=600 height=600>
```

# EXEMPLO 02

000	100	200	300	400	500	600
100	—					
200	/					
300	—					
400						
500						
600						

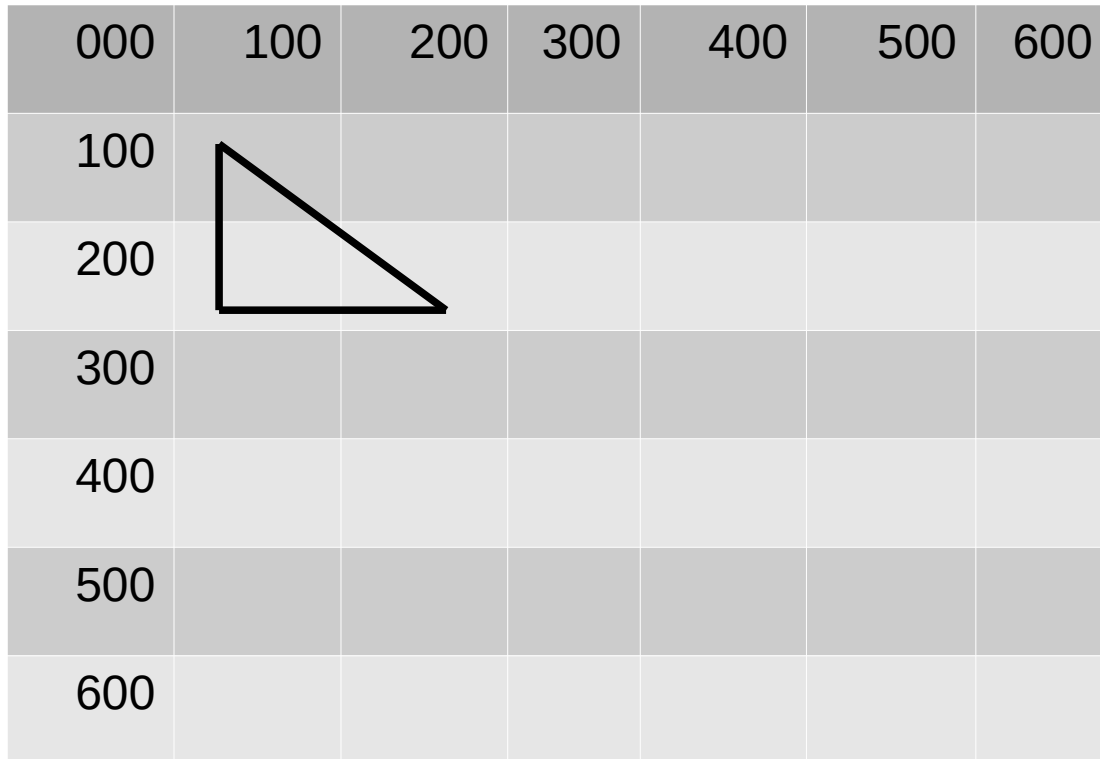
```
ctx.moveTo(100,100); // origem  
ctx.lineTo(200,100); // destino  
ctx.lineTo(100,200); // destino  
ctx.lineTo(200,200); // destino
```

No Caso do Z, os vetores são contínuos, portanto utiliza-se um pontos de origem(**moveTo**) e três pontos de destino(**lineTo**).

## ÁREA DE RENDERIZAÇÃO

```
<canvas id='testeCanvas' width=600 height=600>
```

# EXEMPLO 03



```
ctx.moveTo(100,100); // origem  
ctx.lineTo(200,200); // destino  
ctx.lineTo(100,200); // destino  
ctx.lineTo(100,100); // destino
```

No Caso do triângulo, os vetores são contínuos, portanto utiliza-se um pontos de origem(**moveTo**) e três pontos de destino(**lineTo**).

## ÁREA DE RENDERIZAÇÃO

```
<canvas id='testeCanvas' width=600 height=600>
```

# EXEMPLO DE CÓDIGO COMPLETO

```
<script>
function formasGeometricas() {
    //definindo nome do elemento e contexto das figuras
    var teste = document.getElementById('testeCanvas');
    var ctx = teste.getContext('2d');

    // vetor (reta)
    ctx.beginPath(); // caminho a ser preenchido
    ctx.fillStyle='green'; // cor de preenchimento
    ctx.strokeStyle='blue'; // cor da linha
    ctx.lineWidth = 5; // espessura

    ctx.moveTo(300,100); // origem
    ctx.lineTo(100,200); // destino
    ctx.lineTo(300,300); // destino
    ctx.lineTo(500,200); // destino
    ctx.lineTo(300,100); // destino

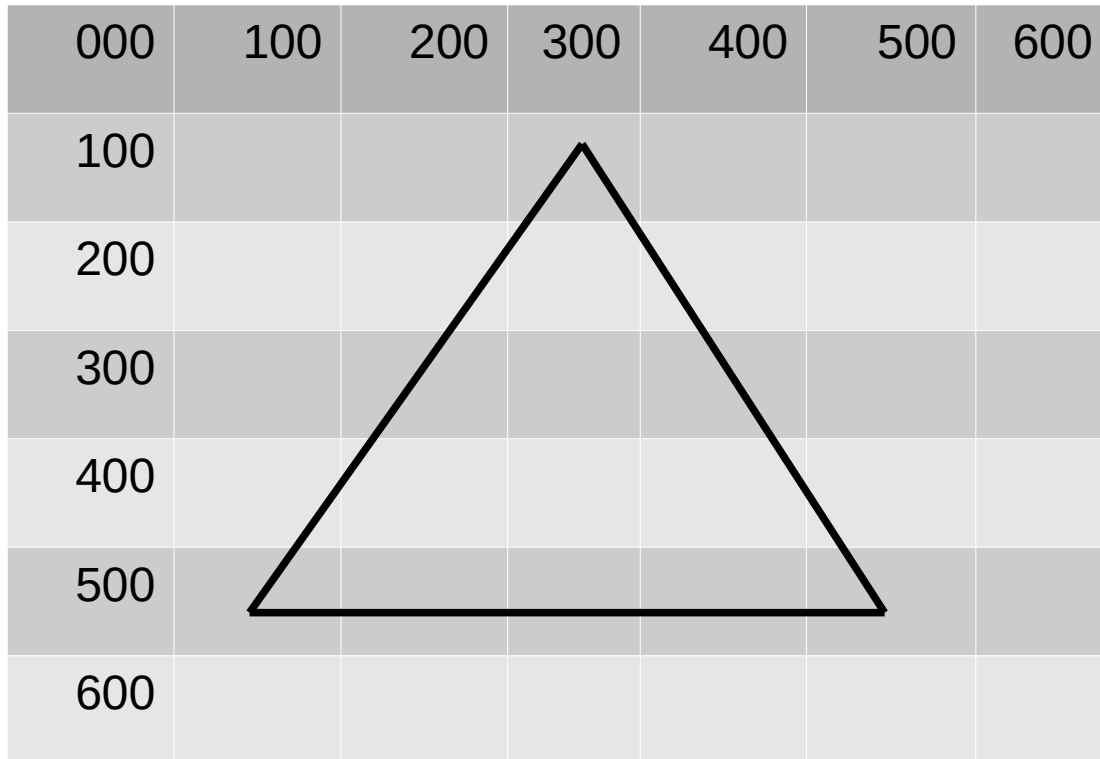
    //ctx.fill(); // preenchimento
    //ctx.closePath(); // finaliza caminho do preenchimento
    ctx.stroke(); // exibe a reta
}
</script>

<center>
<canvas id='testeCanvas' width=600 height=600 style='border:1px solid #000000; padding:15;background:silver;'>
<script>formasGeometricas();</script>
</center>
```

O código acima pode ser copiado para o bloco de notas e salvo como arquivo html para ser executado pelo navegador de internet.

**ATIVIDADES**

# 1) Localize as coordenadas do triângulo abaixo



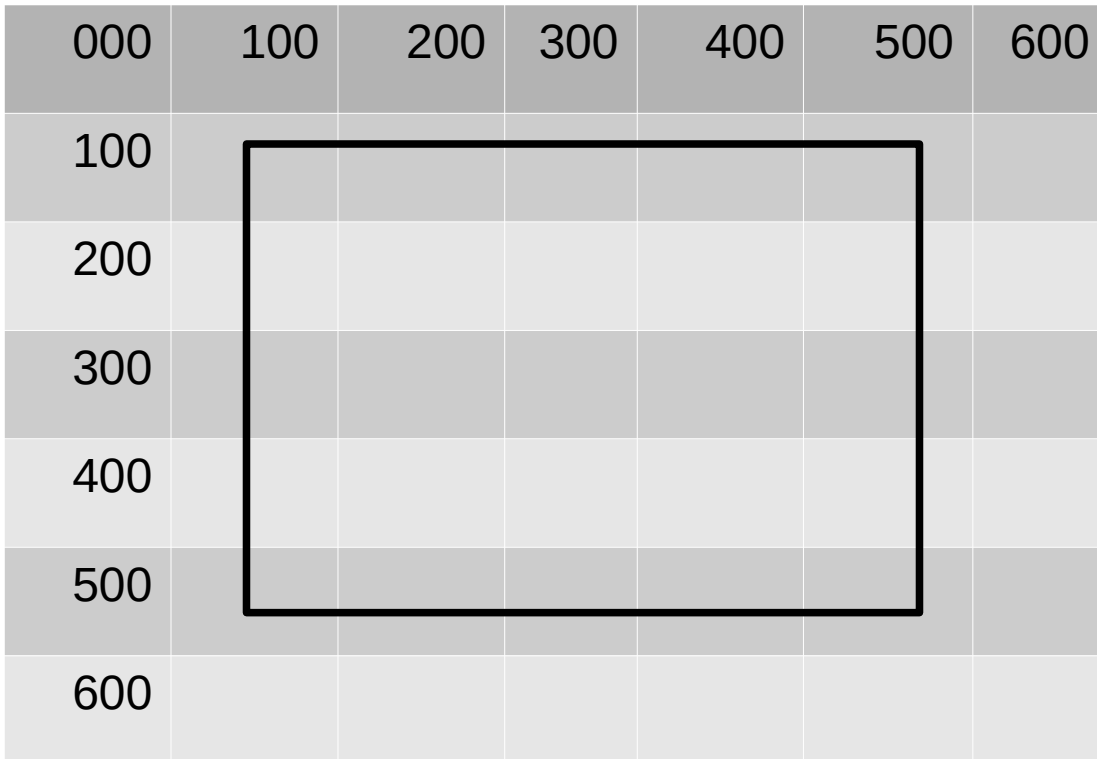
```
ctx.moveTo(); // origem  
ctx.lineTo(); // destino  
ctx.lineTo(); // destino  
ctx.lineTo(); // destino
```

## ÁREA DE RENDERIZAÇÃO

```
<canvas id='testeCanvas' width=600 height=600>
```



## 2) Localize as coordenadas do quadrado abaixo



```
ctx.moveTo(); // origem  
ctx.lineTo(); // destino  
ctx.lineTo(); // destino  
ctx.lineTo(); // destino  
ctx.lineTo(); // destino
```

### ÁREA DE RENDERIZAÇÃO

```
<canvas id='testeCanvas' width=600 height=600>
```

3) Para que serve o comando moveTo()?

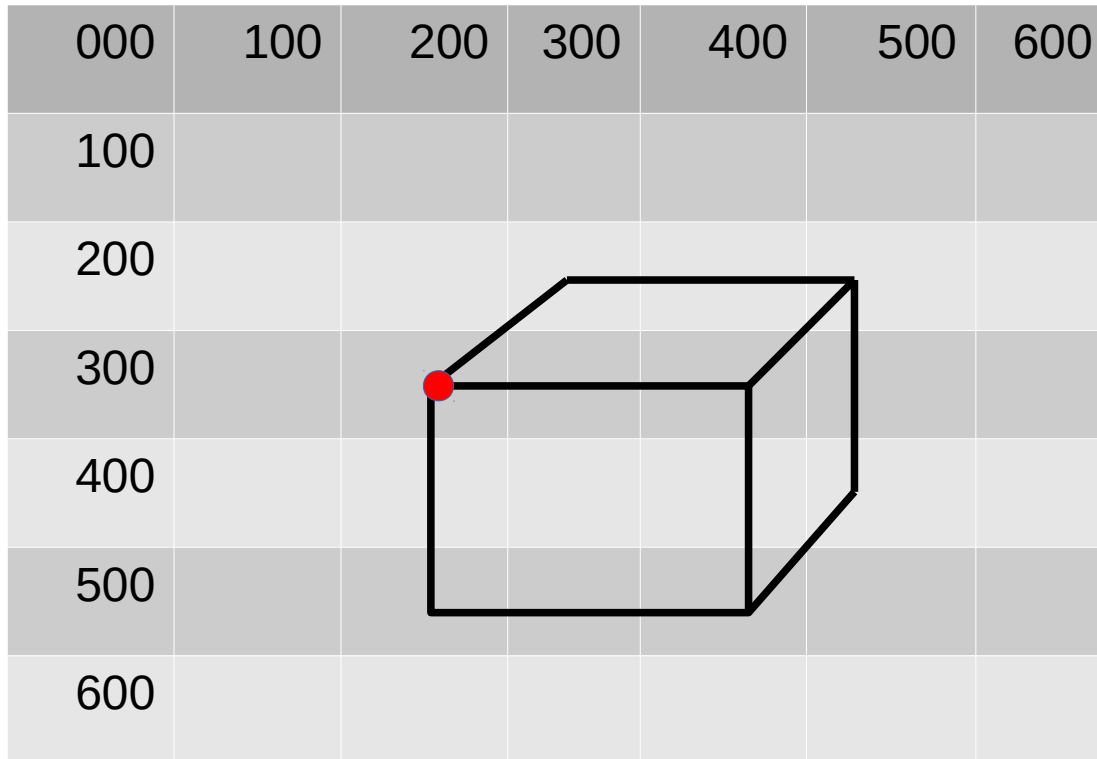
4) Para que serve o comando LineTo()?

5) Para que servem respectivamente os comandos width=600 e height=600?

6) Desenhe as figuras produzidas pelas seguintes coordenadas:

A	B	C
(100,100)	(100,100)	(300,100)
(100,300)	(200,200)	(100,200)
(300,300)	(300,100)	(300,300)
	(400,200)	(500,200)
	(500,100)	(300,100)

## 7) Analise o código abaixo e preencha as lacunas

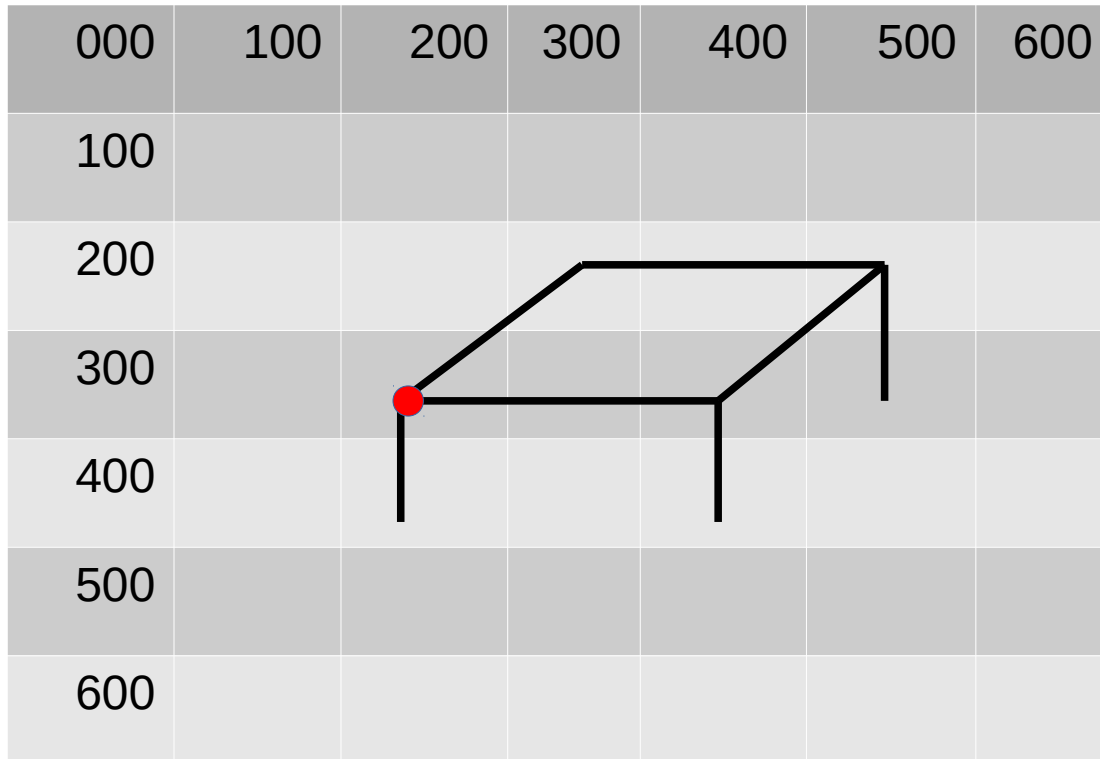


```
ctx.moveTo(200,300); // origem
ctx.lineTo(400,300); // destino
ctx.lineTo(400,___); // destino
ctx.lineTo(200,___); // destino
ctx.lineTo(200,300); // destino
ctx.lineTo(300,___); // destino
ctx.lineTo(500,___); // destino
ctx.lineTo(500,400); // destino
ctx.lineTo(400,___); // destino
ctx.moveTo(400,300); // origem
ctx.lineTo(500,___); // destino
```

ÁREA DE RENDERIZAÇÃO

<canvas id='testeCanvas' width=600 height=600>

## 8) Analise o código abaixo e preencha as lacunas



```
ctx.moveTo(200,300); // origem
ctx.lineTo(300,___); // destino
ctx.lineTo(500,___); // destino
ctx.lineTo(400,___); // destino
ctx.lineTo(200,300); // destino
ctx.lineTo(500,200); // destino
ctx.lineTo(500,___); // destino
ctx.moveTo(500,200); // origem
ctx.lineTo(500,___); // destino
ctx.moveTo(400,300); // origem
ctx.lineTo(400,___); // destino
ctx.moveTo(200,300); // origem
ctx.lineTo(200,___); // destino
```

ÁREA DE RENDERIZAÇÃO

```
<canvas id='testeCanvas' width=600 height=600>
```