

## Sistemas de Coordenadas 2D

O sistema de coordenada no espaço bidimensional (2D) são extremamente importante para o desenvolvimento de games nas *engines* como a **Unity**.

O sistema cartesiano é usado para localizar e representar em um plano, o qual é formado por dois eixos perpendiculares: um horizontal e outro vertical que se cruzam na origem das coordenadas. O eixo horizontal é chamado de abscissa (x) e o vertical de ordenada (y). Os eixos são enumerados compreendendo o conjunto dos **números reais**. Para indicar a localização de um **ponto** no plano, primeiro identifica o ponto no eixo x e em seguida, no eixo y. Por padrão, as coordenadas são representadas pelos pares ordenados (x,y), a Figura 1 traz uma representação de um plano cartesiano.

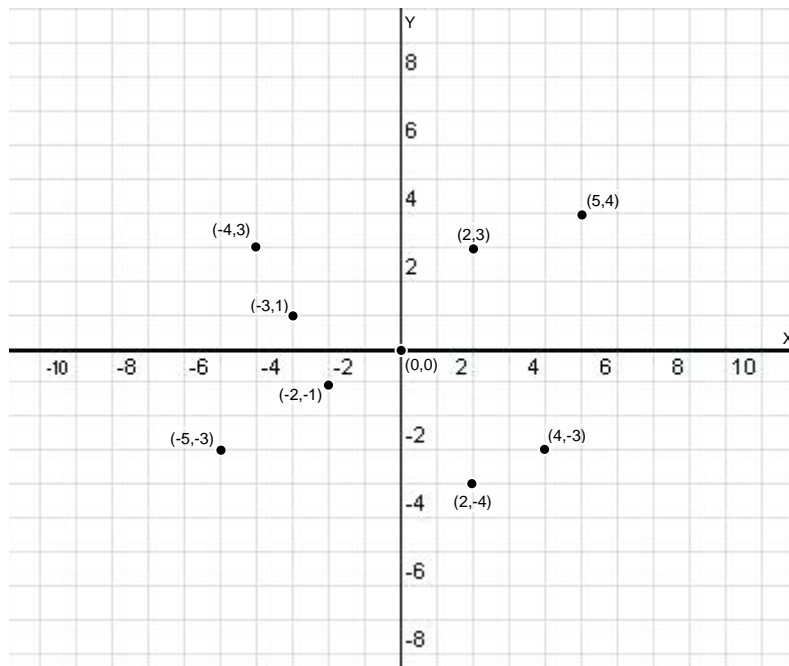


Figura 1: Plano cartesiano

O plano é dividido em 4 partes denominados quadrantes, no **1º quadrante**, a abscissa e a ordenada possui números positivos. Portanto o ponto **(2,3)** está no **1º quadrante**. Então, tendo definido o primeiro quadrante, vamos seguindo o sentido anti-horário para definir os restantes, o ponto **(-3,1)** está no **2º quadrante**, nota-se que a abscissa tem números negativos e a ordenada positivos. No **3º quadrante** temos o ponto **(-2,-1)** onde tanto a abscissa quanto a ordenada possuem números negativos. Por fim, temos o ponto **(4,-3)** que está no **4º quadrante**, onde a abscissa é positiva e a ordenada é negativa, note também que a origem das coordenadas está no ponto **(0,0)**.

## Objetivo do trabalho

Em algumas *engines* é muito comum transladar a origem das coordenadas e assim os quadrantes também são deslocados, o objetivo desse trabalho é escrever um programa que a partir da origem translada (X,Y) do sistema de coordenadas cartesianas e uma quantidade determinada de pontos, informa para cada ponto qual é o seu quadrante transladado.

A entrada do programa é realizada através do teclado seguindo a seguinte configuração:

- No início do programa são lidos a quantidade de pontos (N) e dois números inteiros X e Y representando as coordenadas da **origem translada**.
- Depois é informado um inteiro **N** indicando a quantidade de pontos que serão lidos e processados pelo programa.
- Em seguida serão lidos os **N** pontos pelo programa, para cada ponto lido o programa deve informar, imprimindo no console, qual quadrante translado o ponto pertence, caso o ponto esteja na abscissa transladada ou ordenada transladada o programa imprimirá “origem”.

Ao final da execução, o programa deve imprimir as seguintes informações:

- A **ponto de menor distância** da origem translada e o **valor da distância**.
- A **ponto de maior distância** da origem translada e o **valor da distância**.
- Percentual de pontos por quadrante, sem considerar os pontos que estão na origem (abscissa transladada ou ordenada transladada)

Para Figura 2 abaixo, se o a origem é transladada para **(1,-3)** teríamos o seguinte plano cartesiano:

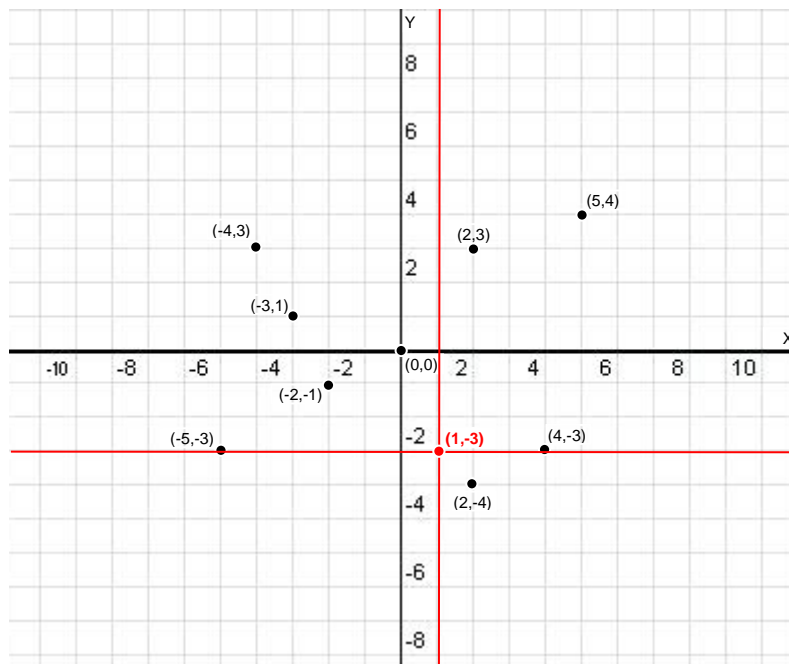


Figura 2: Plano cartesiano com origem translada

E se forem processados os 8 pontos acima com a origem transladada para (1,-3), teríamos a seguinte saída:

```
Ponto ( 5, 4) esta no 1o quadrante.
Ponto ( 2, 3) esta no 1o quadrante.
Ponto (-4, 3) esta no 2o quadrante.
Ponto (-3, 1) esta no 2o quadrante.
Ponto (-2,-1) esta no 2o quadrante.
Ponto (-5,-3) esta na origem.
Ponto ( 4,-3) esta na origem.
Ponto ( 2,-4) esta no 4o quadrante.

Ponto (2,-4) eh o mais proximo, distancia=1,41.
Ponto (5,4) eh o mais distante, distancia=8,06.

Porcentagem de pontos o 1o quadrante 25,00%
Porcentagem de pontos o 2o quadrante 37,50%
Porcentagem de pontos o 3o quadrante 0,00%
Porcentagem de pontos o 4o quadrante 12,50%
```

Na implementação do seu programa você deve usar somente os conceitos vistos no conteúdo das aulas do semestre passado (operadores, tipos, variáveis, estruturas condicionais e de repetição, etc) **não podendo utilizar** estruturas de armazenamento que serão vistas mais adianta no curso, tais como **vetores** e **listas**. Além disso você deve implementar pelo menos duas função, uma para calcular a distância entre uma coordenada X,Y e o ponto divisor, e outra para calcular as porcentagens por região, lembre-se que nas funções não se deve imprimir, somente calcular e retornar.

#### **Observações importantes:**

O programa deve estar bem documentado e implementado na linguagem **Python**. A entrega do trabalho deve ser feita pelo **Moodle (não serão aceitos trabalhos entregues via e-mail)** e será avaliado de acordo com os seguintes critérios:

- Funcionamento do programa;
- O quão fiel é o programa quanto à descrição do enunciado;
- Indentação, comentários e legibilidade do código;
- Clareza na nomenclatura de variáveis;