COLÉGIO PEDRO II – U. E. SÃO CRISTÓVÃO III
**Professor**: João Luiz Lagôas
**Coordenação**: Flávio Costa

## AVALIAÇÃO 02 – (LINGUAGEM DE PROGRAMAÇÃO 02)

## EXERCÍCIO 01 - ANÁLISE DE CÍRCULOS (2,0 PTS)

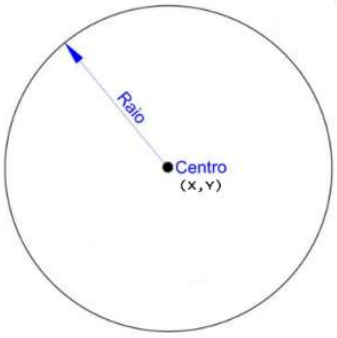
A classe mostrada abaixo é denominada **Ponto.cs**. Ela representa um ponto no plano cartesiano $(x, y)$. Como pode ser visto, esta classe é composta por dois atributos e dois métodos: um construtor e um método estático que calcula a distância entre dois pontos passados como parâmetro.

```
class Ponto
{
    public decimal X;
    public decimal Y;

    public Ponto(decimal x, decimal y)
    {
        X = x;
        Y = y;
    }

    public static decimal Distância(Ponto p1, Ponto p2)
    {
       double dist = Math.Sqrt(Math.Pow(Convert.ToDouble(p1.X - p2).X, 2) + Math.Pow(Convert.ToDouble(p1.Y - p2.Y), 2));
       return Convert.ToDecimal(dist);
    }
}
```

a) Na classe apresentada temos o uso de diversas variáveis: X, Y, x, y, p1, p2 e dist. Classifique cada variável quanto ao seu escopo (de instância, local ou de classe) e quanto ao seu tipo (primitivo ou referência). (0,3 pt)



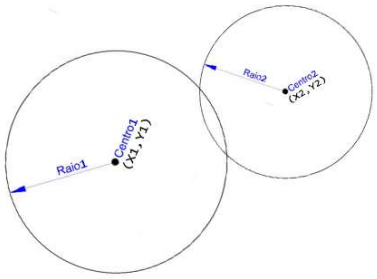

b) Um círculo pode ser desenhado num plano cartesiano a partir de duas informações: um ponto que determina seu centro e um valor positivo que indica o tamanho do seu raio, isto é, a distância do centro até os limites da circunferência. Observe a imagem ao lado. Crie uma classe Círculo contendo dois atributos: um ponto central e um raio. (0,3 pt)

c) Explique se há a necessidade de se aplicar o encapsulamento em algum atributo. Se sim, crie os métodos de acesso e modificação. Justifique a sua resposta. (0,3 pt)

d) Crie um método construtor para essa classe que inicializa seus atributos internos de acordo com os parâmetros passados. (0,3 pt)

e) Crie dois métodos denominados CalcularArea e CalcularPerímetro. Esses métodos são chamados através de uma instância e retornam a área e o perímetro de um círculo, respectivamente. A área é dada por $A = \pi r^2$ e o perímetro é dado por $P = 2\pi r$. (0,3 pt)

f) Crie um método que indique através de uma variável booleana se dois círculos se interceptam, ou seja, se eles se "encostam" em algum ponto. A assinatura do método é dada e a imagem ao lado mostra um exemplo onde dois círculos estão se interceptando. (0,5 pt)




- `public static bool Intercepta(Círculo cir1, Círculo cir2)`

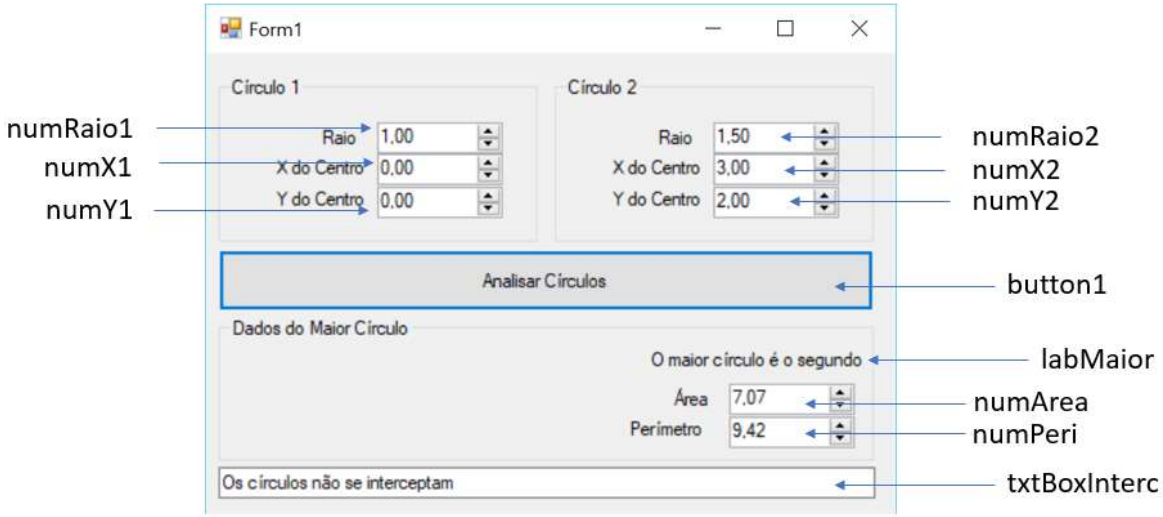
## EXERCÍCIO 02 – USANDO O FORMULÁRIO (2,0 PTS)

Observe o formulário e implemente o método:

- `private void button1_Click(object sender, EventArgs e)`

a) Quando o botão for clicado, os dados presentes em numRaio1, numX1, numY1, numRaio2, numX2 e numY2 deverão ser lidos. Crie duas instâncias de Círculo de acordo com esses valores.
b) O método continuará preenchendo a Label **labMaior** indicando qual círculo é maior (qual apresenta o maior raio) e calculando a sua área e perímetro (preenchendo **numArea** e **numPeri**)
c) Por fim, a Text Box **txtBoxInterc** irá escrever um texto indicando se os círculos lidos se interceptam ou não.

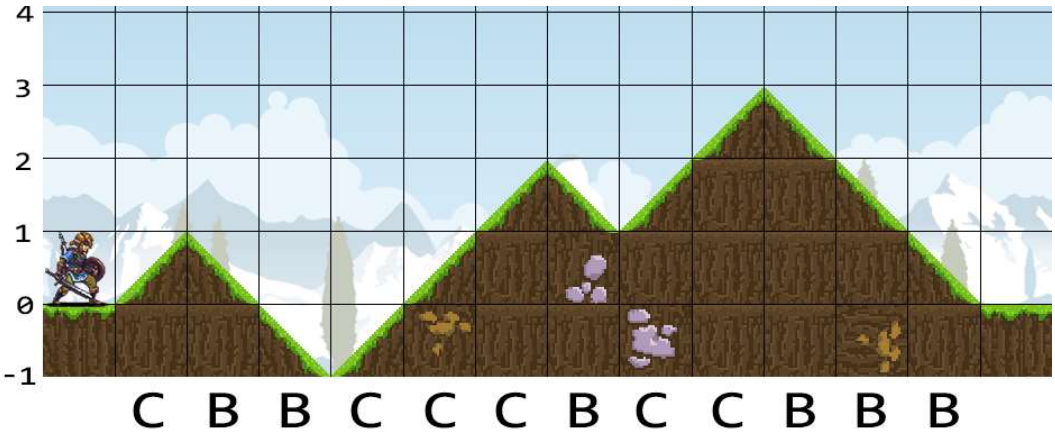
**Nota**: Utilize a classe **Círculo.cs** implementada no exercício anterior.



## EXERCÍCIO 03 - EXPLORANDO O MUNDO (2,0 PTS)

Link é um ávido alpinista e explorador do mundo. Ele rastreia suas escaladas meticulosamente, prestando muita atenção a pequenos detalhes como topografia. Em cada escalada ele faz anotações indicando todos os **passos** que deu: se era um passo para cima, $\boldsymbol{C}$, ou se era um passo para baixo, $\boldsymbol{B}$. Ao término de uma escalada, Link tinha em suas mãos uma sequência de $C$'s e $B$'s, denotando o percurso realizado. Além disso, as escaladas de Link sempre começam no nível do mar e cada passo para cima ou para baixo representa uma mudança de $1$ unidade na altitude.

**Por exemplo**: se o percurso que Link realizou foi tal que $\boldsymbol{percurso} = [\boldsymbol{CBBCCCBCCBBB}]$, então ele alcançou três picos: o primeiro foi alcançado no primeiro passo e tem altitude 1, o segundo foi alcançado no sexto passo e tem altitude 2 e o terceiro foi alcançado no nono passo e tem altitude 3. Além disso, neste exemplo, ele retornou ao nível do mar ao final da escalada. A imagem abaixo ilustra graficamente o percurso que ele realizou.




De acordo com o que foi dito, escreva um programa de Console em C# (a função Main da classe Programa.cs) que faça o seguinte:

a) Leia uma string contendo todos os passos que Link deu. Se a string não respeitar o formato de percurso que foi descrito, isto é, apenas composta por $C$'s e $B$'s, o programa deverá pedir para que o usuário entre com uma nova string que seja válida. (0,4 pt)
b) Seu programa deverá indicar se Link retornou ao nível do mar ou não. Você também deverá indicar o número de passos que ele deu. (0,4 pt)
c) Seu programa deverá indicar quantos picos foram visitados. (0,4 pt)
d) Seu programa deverá indicar qual a altura do pico mais alto. (0,4 pt)
e) Seu programa deverá indicar depois de qual passo Link alcançou um Pico. (0,4 pt)

➢ Você pode realizar as questões b, c, d e e na ordem que julgar melhor.

Um exemplo de entrada e saída é mostrado abaixo:

**Entrada:**
Entre com o percurso:
CBBCCxBaCC
O percurso digitado não é válido!
Entre com o percurso:
CBBCCCBCCBBB

**Saída:**
Link, você retornou ao nível do mar e deu 12 passos.
3 picos foram visitados.
O ponto de altitude mais alto alcançado foi 3.
O 1° pico foi alcançado no passo 1.
O 2° pico foi alcançado no passo 6.
O 3° pico foi alcançado no passo 9.