



C# Avançado

Exercícios Propostos

Operadores e Casting

1 Exercício

Crie uma estrutura `Vector` para representar o conceito de um vetor na matemática. O vetor deve armazenar as coordenadas `X` e `Y` na forma de inteiros. O código base da estrutura é este:

```
public struct Vector
{
    private int x;
    private int y;

    public Vector(int x, int y)
        : this()
    {
        this.x = x;
        this.y = y;
    }
}
```

Nesta estrutura, sobrescreva os operadores `+` e `*`. O operador `+` deve permitir a soma de dois vetores v_1 e v_2 , que resulta em um novo vetor v_r . A soma é feita através da soma das coordenadas, desta forma:

$$v_r = (v_1x + v_2x, v_1y + v_2y)$$

Já o operador `*` deve permitir a multiplicação de um vetor v por um número inteiro m . O resultado é um vetor v_r , cujas coordenadas são multiplicadas por este número, desta forma:

$$v_r = (vx * m, vy * m)$$

Para testar seu código, utilize a sequência de execução abaixo, que define dois vetores, soma ambos e multiplica o vetor resultante por 3:

```
Vector v1 = new Vector(2, 3);
Vector v2 = new Vector(4, 5);
Vector v3 = v1 + v2;
Vector v4 = v3 * 3;
```

O resultado final é um vetor cujas coordenadas são (18, 24).

2 Exercício

Tomando como base a mesma estrutura criada no Exercício 1, crie um casting customizado que permita converter, de forma implícita, um `Vetor` em um objeto `string`. Esta string deve representar o vetor no formato `(x, y)`.

Crie também um indexador que permita extrair ou atribuir valores às coordenadas `x` e `y` do vetor. Os caracteres `'X'` ou `'Y'` devem ser usados como indexadores. Veja um exemplo:

```
Vector v = new Vector();  
v['X'] = 5;  
v['Y'] = 7;  
int x = v['X'];  
int y = v['Y'];
```