

**Programação Orientada a Objeto**

**Objetivo**

Aplicar o conhecimento de herança em um caso prático.

**Exercício**

****

Imagine que você está ajudando na implementação do jogo de nave Arcade, e você precisa implementar a lógica dos elementos do jogo: nave, nave inimiga, asteróide. Eles são todos objetos gráficos que ocupam uma posição no espaço (x, y) e à medida que se movem ocupam uma nova localização, As naves têm uma velocidade e uma direção. A direção pode ser ‘N’, 'S', 'L', 'O', (sul, norte, leste ou oeste). Os asteroides podem ir para qualquer posição, independentemente de sua direção atual.

O problema é para resolver o movimento, não é necessário resolver outra operação. Embora, para usar no futuro, seja necessário sobrescrever toString, hashCode e equals, tanto na nave quanto no asteroide. Para isto, as seguintes classes foram projetadas. Como seria sua implementação em Java? Crie o código para as classes abaixo no seu **IntelliJ**.

|  | **Objeto** |  |
| --- | --- | --- |
|  | * posx: int * posy: int * direcao:char |  |
|  | * Objeto(x:int, y:int, direcao:char) * irA(x:int, y:int, direcao:char) |  |



| **Nave** |  | **Asteroide** |
| --- | --- | --- |
| * velocidade:double * vida:int |  | * danos:int |
| * Nave( x:int, y:int, direcao:char, velocidade:double) * irA(x:int,y:int, direcao:char) * girar(direcao:char) * restaVida(int valor) |  | * Asteroide( x:int, y:int, direcao:char, danos:int) * irA(x:int,y:int, direcao:char) |