

Naruto - Batalha Ninja



Para quem não sabe, Naruto é uma série de anime e mangá criada por Masashi Kishimoto que recebeu adaptação para anime em 2002. A trama gira em torno de Naruto Uzumaki, um jovem Ninja que constantemente busca por reconhecimento e tem o sonho de se tornar um Hokage, o líder máximo e mais poderoso de sua vila. Durante a história, ocorreram diversas batalhas Ninjas em prol de algum objetivo, tais como: exames para subir de nível, guerras, missões, entre outras.

Neste exercício, vamos modelar e simular uma batalha entre dois ninjas. Leia atentamente as regras e transforme tudo o que está sendo solicitado nas aulas Java respeitando os princípios da OO que vimos até aqui.

Jutsu

Jutsu (術; Literalmente significa "habilidades/técnicas") são as artes místicas que um Ninja utiliza na batalha. Para usar uma técnica, o Ninja precisará usar seu chakra. Neste exercício, vamos considerar o chakra como a energia que o Ninja gasta para se manter vivo e atacando.

Modelagem

1. Modele uma classe de Jutsu que possui como atributos a quantidade de chakra que consome quando utilizado e o dano (quantidade de chakra que abate o oponente).

2. Todos os atributos serão informados no momento em que o Jutsu for instanciado e não poderá ser alterado posteriormente.
3. Sobre o atributo que se refere à quantidade de chakra consumido pelo Jutsu, ele deve ser do tipo `int` e valor 5 deve ser o máximo que pode ser atribuído.
4. Sobre o atributo que se refere ao dano (quantidade de chakra que o Jutsu remove do oponente), ele deve ser do tipo `int` e valor 10 deve ser o máximo que pode ser atribuído.

Ninja

Ninjas é o foco principal e a principal potência militar da série. A maioria vem de uma aldeia oculta, alguns dos quais vêm de clãs Ninja especializados, e irão realizar as batalhas do nosso exercício.

Modelagem

1. Modele uma classe Ninja que possui como atributos o nome, o chakra que possui e o seu Jutsu principal.
2. Todos os atributos serão informados no momento em que o Ninja for instanciado.
3. O chakra do Ninja deve ser do tipo `int` e inicializar com valor 100.
4. O Ninja deve possuir um método `atacar` que receba como parâmetro o seu oponente Ninja.
5. O Ninja deve possuir um método `receberGolpe` que receba por parâmetro um valor do tipo `int` correspondente ao dano do golpe recebido.
6. O método `atacar` deve invocar o método `receberGolpe` do oponente, informando o dano do Jutsu principal do Ninja.
7. O método `receberGolpe` deve diminuir o chakra do Ninja o valor do dano recebido.
8. Quando atacar um oponente, o valor do chakra deve ser atualizado de acordo com o consumo do Jutsu.

Batalha

1. Modele uma classe de Batalha em que o **único** método público deverá ser o `lutar` que receberá como parâmetro os dois ninjas da luta. Segue exemplo de assinatura do método: `public Ninja lutar(Ninja primeiroNinja, Ninja segundoNinja)`.

2. No método `lutar`, cada Ninja ataca seu oponente 3 vezes.
3. O método `lutar` deve retornar como Ninja vencedor aquele que possui o maior chakra ao final da luta.
4. Se houver empate, o Ninja do primeiro parâmetro deverá ser retornado como vencedor.

Regras Gerais

Uchiha Itachi



Se um dos Ninjas se chama **Itachi** , ele deve vencer sempre, independentemente do oponente.
Obs: não ouse afirmar que ele não é o melhor de todos.

Exemplos de teste

```
@Test
public void deveRetornarNinjaComJutsuMaisForteSeOsDoisGastamOMesmoChakra() {
    Jutsu jutsuNinjaUm = new Jutsu(5, 10);
    Ninja ninjaUm = new Ninja("Naruto", jutsuNinjaUm);

    Jutsu jutsuNinjaDois = new Jutsu(5, 8);
    Ninja ninjaDois = new Ninja("Gaara", jutsuNinjaDois);

    Batalha batalha = new Batalha();

    Ninja ninjaVencedor = batalha.lutar(ninjaUm, ninjaDois);

    Assert.assertSame(ninjaUm, ninjaVencedor);
}
```

```
@Test
public void
deveAtualizarOChakraDoOponenteDeAcordoComODanoDoJutsoQuandoAtacar() {
    Jutsu jutsuNinjaAtacante = new Jutsu(5, 10);
    Ninja ninjaAtacante = new Ninja("Naruto", jutsuNinjaAtacante);

    Jutsu jutsuNinjaOponente = new Jutsu(5, 8);
    Ninja ninjaOponente = new Ninja("Gaara", jutsuNinjaOponente);

    int nivelChakraEsperado = 90;

    ninjaAtacante.atacar(ninjaOponente);

    Assert.assertEquals(nivelChakraEsperado, ninjaOponente.getChakra());
}
```

Referências

[Sobre a série](#)

[Chakra](#)

[Jutsu](#)

Testes Obrigatórios

- Devem existir **obrigatoriamente** na solução entregue testes unitários com os seguintes nomes:
 - deveRetornarNinjaComJutsuMaisForteSeOsDoisGastamOMesmoChakra
 - deveAtualizarOChakraDoOponenteDeAcordoComODanoDoJutsoQuandoAtacar
 - deveRetornarPrimeiroNinjaComoVencedorQuandoONomeForItachi
 - deveRetornarSegundoNinjaComoVencedorQuandoONomeForItachi
 - deveRetornarPrimeiroNinjaComoVencedorQuandoEmpatar